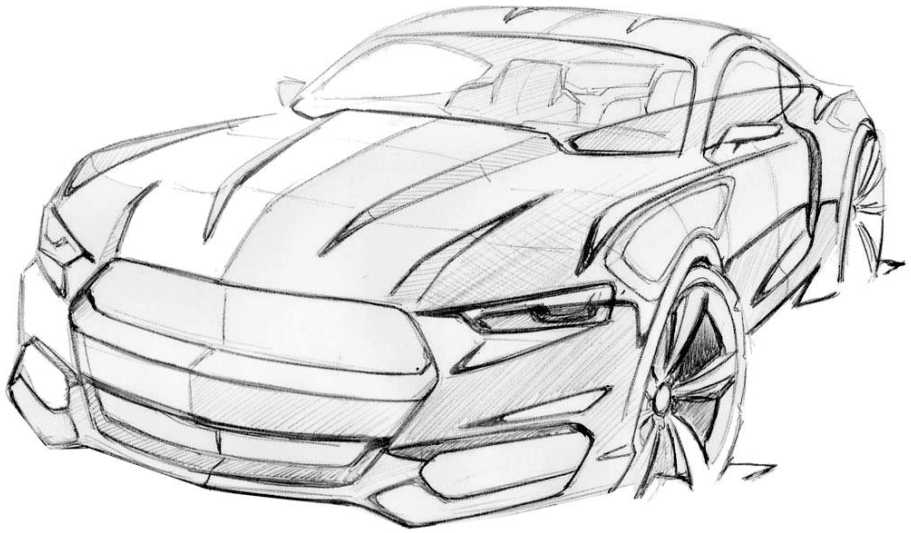
2015

Análisis de datos aplicado a siniestros de automóviles



Eduardo González González

Universidad Carlos III de Madrid

TRABAJO DE FIN DE GRADO

09/06/2015

Agradecimientos

Me gustaría agradecer en primer lugar a mi tutor del proyecto, Antonio Berlanga, su gran implicación y total disponibilidad cuando he necesitado de su consejo. Sin su experta dirección no habría sido posible llevar a buen puerto este trabajo.

Mi más sincero agradecimiento a la Carlos III, al profesorado que durante todos estos años me ha impartido clase y a las personas involucradas en el buen funcionamiento de la universidad, por brindarme la oportunidad de desarrollarme académica y personalmente, por abrirme las puertas a una nueva visión de las cosas.

Quiero darles las gracias a mis compañeros durante todos estos años, por echarme un cable cuando me hizo falta, por las largas horas de estudio juntos y por haber compartido conmigo una etapa tan importante.

También quiero agradecer de corazón a mis amigos más cercanos, a mis hermanos. Ellos han hecho más liviana la carga durante todos estos años, me han dado un motivo para seguir en los malos momentos.

Por último, y más importante, me gustaría agradecer a mis padres todo su apoyo, porque ellos han sido el verdadero pilar en esta empresa, desde el primer al último minuto y con total entrega. A ellos les dedico este proyecto.

Abstract

Dada la creciente importancia que está tomando en la actualidad la lucha contra el fraude en el seguro del automóvil, es necesario desarrollar nuevas herramientas de evaluación que complementen a los métodos tradicionales de peritación.

Este proyecto tiene como objetivo realizar un modelado de las características esenciales que presenta un siniestro de automóvil, como su tipología, dirección de impacto o grado de severidad, con el fin de automatizar el proceso de tasación y facilitar la detección de valoraciones materiales fraudulentas o erróneas. Para este procedimiento se tendrán en cuenta informes periciales de accidentes pasados proporcionados por una importante compañía española de gestión de siniestros.

Durante el desarrollo de este trabajo se utilizarán diversas técnicas de *data mining* y aprendizaje automático, tanto supervisado como no supervisado, sobre un fichero de datos que contiene vectores de entrada con ejemplos de valoraciones. De la experimentación se extraerán diversos modelos de conocimiento que servirán al propósito general del proyecto.

Palabras clave

siniestros de automóviles, peritación, fraude, análisis de datos, data mining, aprendizaje automático, aprendizaje supervisado, aprendizaje no supervisado, clustering.

Índice

Índice de Ilustraciones

Índice de Tablas

Índice de Gráficas

1. Introducción

1.1.Contexto, objetivos y motivación

A día de hoy, en Internet se generan, cada segundo, monumentales cantidades de datos procedentes de todas las actividades humanas imaginables, desde el paso de un vehículo por una carretera al parte meteorológico por hora en cada uno de los núcleos urbanos de un país, pasando por las fotos que cada uno cuelga en su página personal de una red social o las últimas tendencias en moda de los famosos. En esta que llaman la Era de la Información, existe tal cantidad de la misma alojada en servidores a lo largo y ancho del globo que se ha hecho indispensable diseñar y planificar todo tipo de tecnologías para el manejo de semejante cantidad de datos.

A su vez, en los últimos años, la conciencia sobre el valor de la información ha crecido, teniendo su máximo exponente en el ámbito empresarial y de los negocios, donde la información es considerada un bien patrimonial con capacidad de ser explotado, apreciado pero a la vez peligroso si se descuida o no se maneja con cuidado. La información puede ayudar a la hora de tomar decisiones, optimizar, predecir o planificar en un ámbito determinado, obteniendo una ventaja que puede traducirse en un beneficio económico.

El desarrollo de la tecnología actual ha permitido disponer de potentes herramientas para el manejo de los datos. A través de disciplinas como la Estadística, la Inteligencia Artificial, el Aprendizaje Automático o la Minería de Datos, puede extraerse valioso conocimiento que servirá a posteriori como catalizador de un mejor rendimiento.

Por otra parte, el sector del automóvil es siempre uno de los motores económicos en aquellos países en los que se producen vehículos, debido al nivel de dependencia de la automoción que ha alcanzado el ser humano en todas sus actividades cotidianas. El mercado del automóvil español se encuentra entre los 5 más potentes de Europa, junto a Alemania, Reino Unido, Francia e Italia. Semejante volumen de negocio no sólo mueve dinero en el ámbito de las ventas de vehículos nuevos, sino también en reventa de vehículos de segunda mano, piezas, reparaciones, seguros y mantenimiento de aquellos vehículos que componen el parque móvil.

En el gremio de las aseguradoras de automóviles, disponer de información fiable y veraz se considera vital a la hora de realizar valoraciones en siniestros, peritaciones y análisis de casos de fraude. Una valoración incorrecta afecta tanto a la compañía como al resto de asegurados, con lo que es deseable obtener resultados óptimos en el análisis de los informes periciales. En la actualidad, el fraude al seguro del automóvil constituye más del 70% de todos los casos de estafas a compañías aseguradoras en España. Por ello, se considera una de las áreas del negocio más delicadas, con un espectacular aumento de inversión en los últimos años.

Los nuevos modelos de seguros, con infinidad de modalidades de pólizas, coberturas y otras combinaciones, añadidos al colosal tamaño de las bases de datos de clientes actuales, requieren de nuevas técnicas y herramientas para el manejo de toda esa información. En este sentido, en el día a día los peritos se valen de diversas herramientas informáticas para realizar su trabajo, entre las que se

incluyen avanzadas aplicaciones para el análisis de siniestros, estándares de codificación, baremos o servicios de consultoría desde otras entidades.

El componente humano es sin duda el eslabón más débil de la cadena de peritación. Los riesgos atribuibles a las personas no solo dependen de la honestidad a la hora de realizar partes o evaluar desperfectos, sino que pueden producirse también debido a la necesidad de realizar un gran número de tareas repetitivas que pueden conducir a error, o de la aparición de nuevas técnicas de fraude desconocidas por los técnicos. La tecnología puede conseguir reducir estos riesgos introduciendo progresivamente un mayor número de fases automatizadas, aumentando el control.

Este proyecto pretende aportar nuevas herramientas y soluciones para la peritación y valoración de desperfectos, considerada como la fase menos "automatizable" del proceso de gestión de un siniestro, debido a su complejidad.

Complementando la finalidad principal, se plantean los siguientes objetivos específicos:

- Realizar un estudio de las características esenciales que presenta un siniestro de automóvil, teniendo en cuenta informes de desperfectos y reparaciones de percances pasados.
- Diseñar un patrón concreto que permita catalogar valoraciones periciales atendiendo al grado de severidad del siniestro que evalúan.
- Conformar un modelo de auditoría de daños relacionando las reparaciones más típicas presentes en los informes con los tipos de impacto más comunes en un vehículo.

Este proyecto no persigue el diseño de un mecanismo clasificador de valoraciones fraudulentas, sino un mero evaluador de las características esenciales de un siniestro, con el fin de aportar información para que terceras partes tomen la decisión.

2. Estado del arte

2.1.Peritación de siniestros de automóviles

Tradicionalmente, la labor de valoración de siniestros para el mercado de postventa de automoción la ha llevado a cabo la figura del perito o ingeniero técnico [1], una persona encargada de evaluar los daños que presenta un automóvil y emitir un informe que será utilizado para establecer la cuantía económica de la compensación. Para esta indemnización se suelen cotejar registros pasados y se tienen en cuenta otros criterios establecidos por la compañía aseguradora para la que trabaja el perito.

El perito ejerce de representante de la compañía de seguros ante el taller donde se realizarán las reparaciones pertinentes, esto es, el mismo ente que paga la nómina del perito es el que además pagará la factura del taller [2].

Ilustración 1: Perito valorando un siniestro

Atendiendo a esta afirmación, las responsabilidades de ambas partes implicadas se definen, de una manera objetiva, de la siguiente manera:

- Perito: dispone de un título académico que le cualifica para determinar qué piezas o partes del vehículo deben ser cambiadas o reparadas, realizando las estimaciones pertinentes de costes de mano de obra, costes de reparación, sustitución, chapa o pintura.
- Taller: se encarga de establecer el precio del trabajo que ha determinado el perito en su informe.

Y es en esta situación cuando se pueden producir opiniones enfrentadas, dado que entre la aseguradora y el taller existen intereses opuestos. A la aseguradora le interesa emitir informes con el mínimo gasto posible y al taller le interesa obtener el máximo beneficio de su trabajo.

Para remediar esta situación, muchas aseguradoras trabajan exclusivamente con talleres concertados y dotan a sus peritos de herramientas y procedimientos que garanticen la fiabilidad y objetividad en los siniestros analizados.

Además de las posibles desavenencias que puedan existir entre aseguradora y taller, el cliente (conductor o asegurado) puede estar a su vez en desacuerdo con la valoración realizada por el perito de la empresa. La ley ampara que el cliente pueda contratar a su propio perito, y la valoración de éste será tan válida como la presentada por el primero. Como estipula la ley 50/1980 del contrato de Seguro [3], en caso de no llegar a acuerdo ambos técnicos, la decisión quedará a expensas de un tercero designado por ambas partes (cliente y aseguradora) o por un juez.

A continuación se enumeran las plataformas más comunes utilizadas para la peritación en el sector de automoción.

2.1.1. Audatex

La mayoría de peritos trabajan con diversas herramientas de referencia para las valoraciones, pero entre ellas la más destacada es Audatex, considerada el estándar del mercado para el peritaje. Audatex ofrece una amplia gama de servicios destinados a compañías aseguradoras, gabinetes periciales, compañías de *renting* y garantías mecánicas, redes de talleres franquiciados y talleres particulares.

Ilustración 2: Logotipo de Audatex

Según establece la compañía en su web [4], los servicios ofertados son:

- Su aplicación insignia, AudaPlus (Ilustración 3), considerada como la "solución estándar en el mercado, con las mejores funcionalidades de reparación y valoración diseñada por y para profesionales". AudaPlus incluye gran cantidad de módulos, *plugins* y extensiones con otras aplicaciones de la compañía.
- Una Base de Datos con el 99% de modelos presentes en el parque móvil español actual, que incluye desde turismos a motos, pasando por todoterreno y vehículo industrial ligero/pesado.
- Servicios de mantenimiento tanto preventivo como correctivo.
- "Gráficos de captura inteligente" que identifican vehículos mediante una innovadora tecnología.
- Información detallada en Carrocería y mecánica.
- Herramienta específica para lunas.
- Diversas funciones y herramientas como AudaVin (identificación de vehículos), AudaCheck (ayuda para verificación y validación de siniestros), AudaEstadísticas (informes estadísticos), AudaSubastas (gestión de restos de siniestros).
- Integración de la plataforma con otros sistemas de gestión.

Audatex y su aplicación AudaPlus son sin duda el referente en la peritación de automoción en este país, habiéndose situado en los últimos años por delante de sus competidores. Sin duda sus aplicaciones son sencillas de utilizar, potentes y vistosas.

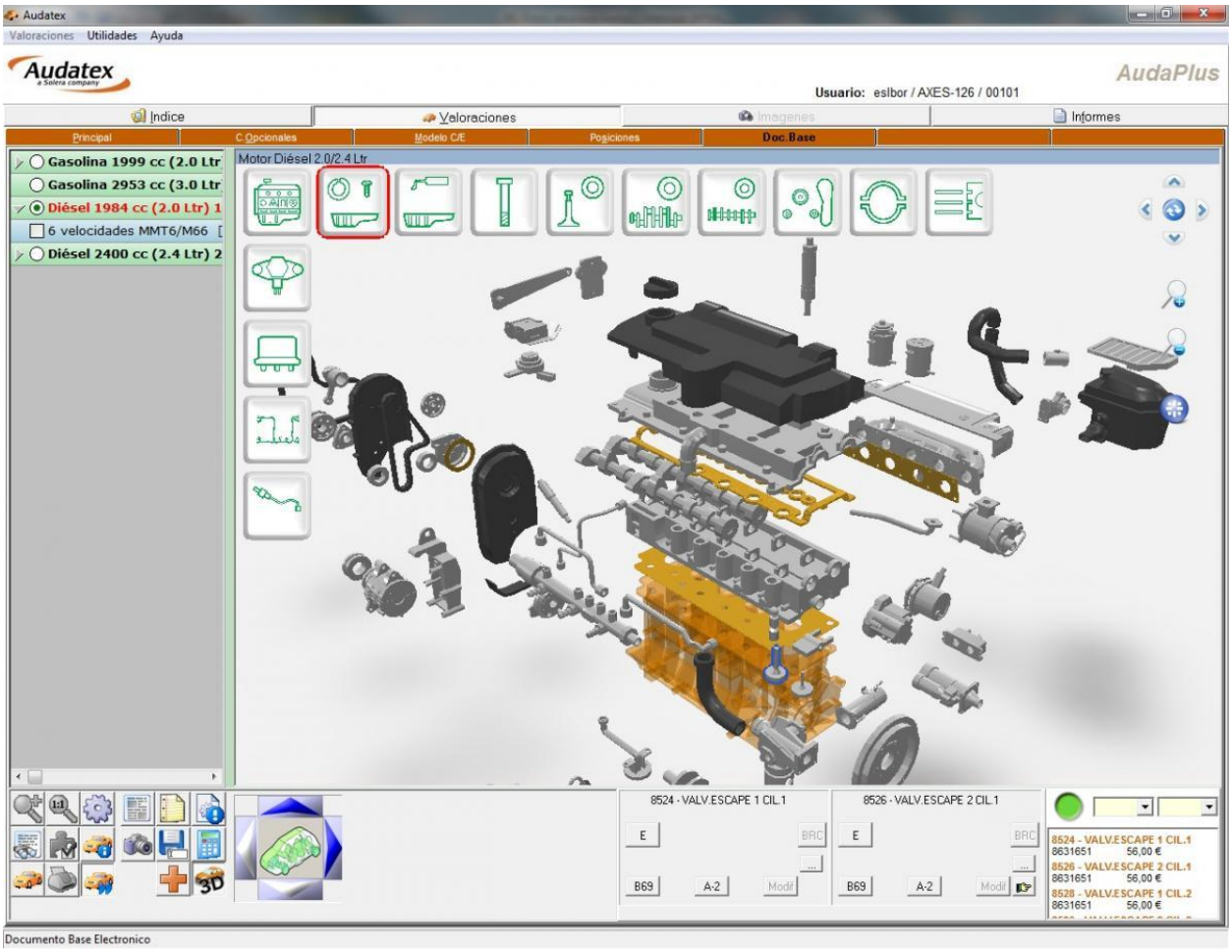


Ilustración 3: Captura de pantalla de AudaPlus

Sin embargo, el trabajo de perito no puede reducirse exclusivamente a una única plataforma, dado que existen pequeñas funcionalidades o baremos y referencias que sólo se encuentran disponibles en una u otra ubicación. Es por ello que existen otras plataformas bastante conocidas a disposición de los técnicos.

2.1.2. GT Motive Estimate

Otra de las aplicaciones más populares del mercado es GT Motive Estimate, perteneciente al grupo empresarial español GT Motive (Einsa) y a la multinacional estadounidense Mitchell International, líder del sector Norteamericano [5].

Ilustración 4: Logotipo de GT Motive

Esta solución también consiste en una aplicación que informatiza y virtualiza la labor tradicional del perito, permitiendo realizar una valoración más precisa y adaptada a los estándares del mercado. Según proclama en su web, GT Motive Estimate es un software de valoración de siniestros que permite calcular el coste de las reparaciones llevadas a cabo en un vehículo por daños de colisión, averías mecánicas y servicios de inspección. La aplicación (ver Ilustración 5) proporciona información relativa a precios y referencias de piezas, tiempos de mano de obra y materiales.

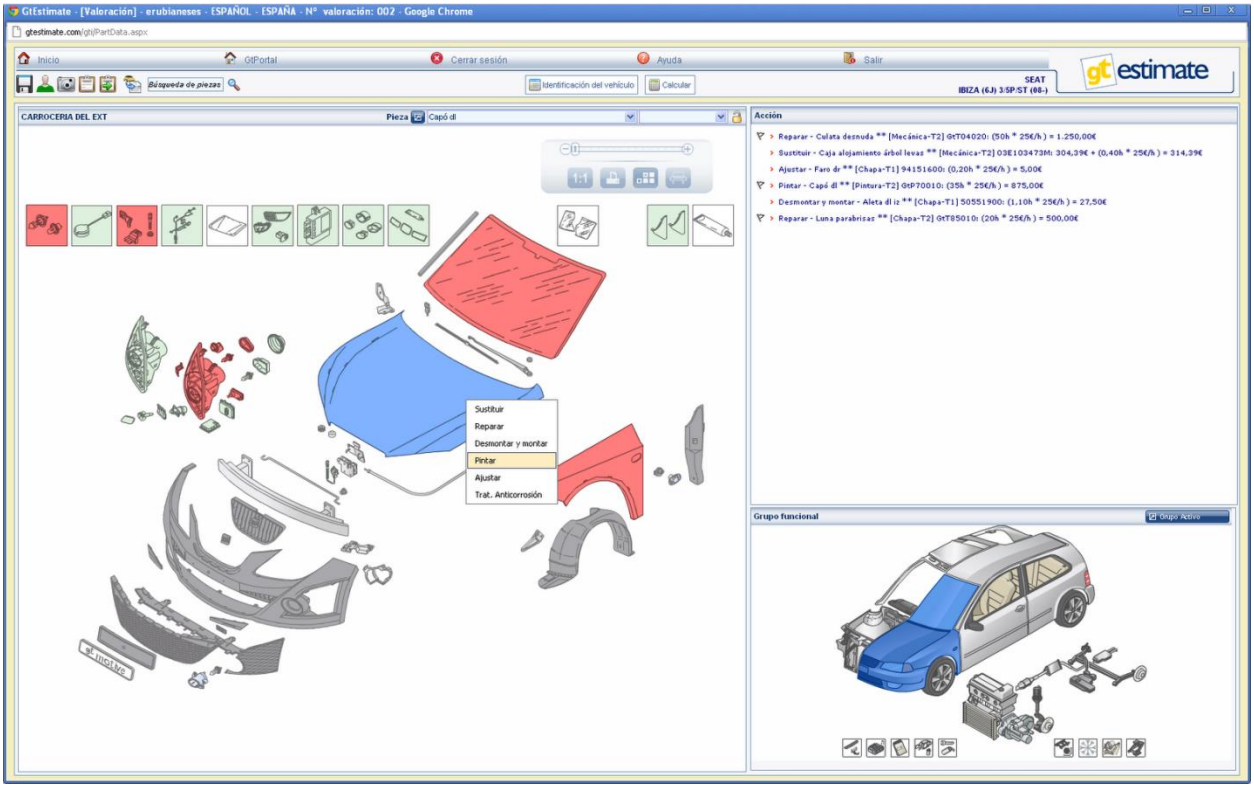


Ilustración 5: Captura de pantalla de GT Estimate

2.1.3. GANVAM

Se denomina GANVAM a la Asociación Nacional de Vendedores de Vehículos a Motor, Reparación y Recambios. Fundada en 1957, se considera la organización más antigua y representativa del mercado de la distribución y la reparación en España. GANVAM carece de dependencias con otros estamentos y por tanto es considerada neutral, actuando con plena y total libertad en interés de sus asociados.

Ilustración 6: Logotipo de GANVAM

Como establece en su página web [6], GANVAM tiene por objetivos "La representación, asesoramiento y defensa de los legítimos intereses de sus afiliados, en el ejercicio de su profesión", además de "La relación y colaboración con los diferentes departamentos de la Administración: Estatal, Autonómica y Municipal, Asociaciones de Consumidores y de Profesionales, ante los que GANVAM está considerada como un interlocutor válido y cualificado para participar en todos aquellos asuntos que afectan profesionalmente a sus asociados".

El principal motivo por el cual muchos peritos utilizan los servicios de GANVAM es porque dispone de una de las más importantes bases de datos de información relacionada con el sector, considerada una referencia a nivel nacional, que incluye:

- Valoraciones estadísticas del mercado de Vehículo de Ocasión; todo terrenos, industrial, motocicletas, tractores, etcétera.
- Extensos informes de estadísticas sobre matriculación de vehículos nuevos, ocasión, bajas y transferencias entre propietarios.
- Informes y estudios de mercado de periodicidad mensual, trimestral y anual.
- Baremos y tablas de referencia.
- Información sobre jornadas, foros y ferias de automoción a lo largo de toda la geografía española.

2.1.4. SINCO

La empresa Tecnologías de la Información y Redes para las Entidades Aseguradoras S.A. (TIREA) pone a disposición de los profesionales del sector del seguro automovilístico el SINCO, o Fichero Histórico del Seguro del Automóvil.

Ilustración 7: Logotipo del SINCO (Fichero Histórico de Seguros de Automóviles)

Según la web de TIREA [7], dicho servicio consiste en la posibilidad de acceder de manera inmediata al historial de seguros actualizado de cualquier empresa aseguradora en el momento de tarificar una póliza. De esta manera, se satisfacen los objetivos siguientes:

- Tarificar adecuadamente los riesgos en función de cada tomador del seguro.
- Promover la transparencia del mercado del seguro del automóvil de modo que los asegurados tengan un mayor acceso al conjunto de ofertas del sector, pudiendo así buscar la oferta que mejor se adecúe a sus necesidades.
- Establecer un sistema ágil, seguro e imparcial, que proporcione la información que demandan las Entidades Aseguradoras.

El SINCO está compuesto fundamentalmente por una base de datos sectorial con el historial completo y actualizado de cada contratante de seguro de automóviles, por lo que se considera una importante referencia a la hora de realizar el trabajo de peritaje.

2.1.5. Eurotax

Otro de los referentes más importantes para la consulta de tasaciones, especificaciones de vehículos y datos del mercado es Eurotax, el observador independiente del mercado de automoción a nivel europeo más importante, con una trayectoria de más de 80 años [8].

Ilustración 8: Logotipo de Eurotax

Eurotax ofrece a sus clientes una base de datos del parque móvil europeo con más de 90.000 modelos de vehículos y equipamientos, permitiendo identificar inequívocamente cada uno de ellos mediante un estándar de codificación ampliamente extendido dentro del sector. A día de hoy, esta base de datos paneuropea está considerada como la más completa del mercado.

Por otro lado, Eurotax también ofrece estimaciones de valores de venta en Vehículo de Ocasión, poniendo a disposición de los clientes un potente motor estadístico alimentado por millones de observaciones del mercado. Este procedimiento genera un modelo actualizado de valor de venta completo, fiable y crítico para la toma de decisiones.

Eurotax ofrece además, los siguientes servicios y soluciones de alto valor añadido:

- Valoraciones, datos técnicos y datos de administración de flotas.
- Estimaciones de daños.
- Sistemas de administración de Vehículo de Ocasión y para talleres.
- Servicios web integrados.

2.2.Detección de fraude

Desde los propios orígenes del concepto de compañía aseguradora, allá por el siglo XIX, se vienen registrando comportamientos fraudulentos en el mercado de automoción, no se trata por tanto de un fenómeno surgido en los últimos años, sino algo prácticamente implícito a los seguros a lo largo de su historia.

Según el CECAS (Centro de Estudios del Consejo General de los Colegios Mediadores de Seguros) [9], se define el fraude en el seguro como:

> *"Toda actuación de mala fe llevada a cabo por una persona con el objeto de obtener para sí misma, o en beneficio de un tercero, un enriquecimiento injusto e ilícito a expensas de una compañía de seguros, mediante la utilización de un artificio o engaño".*

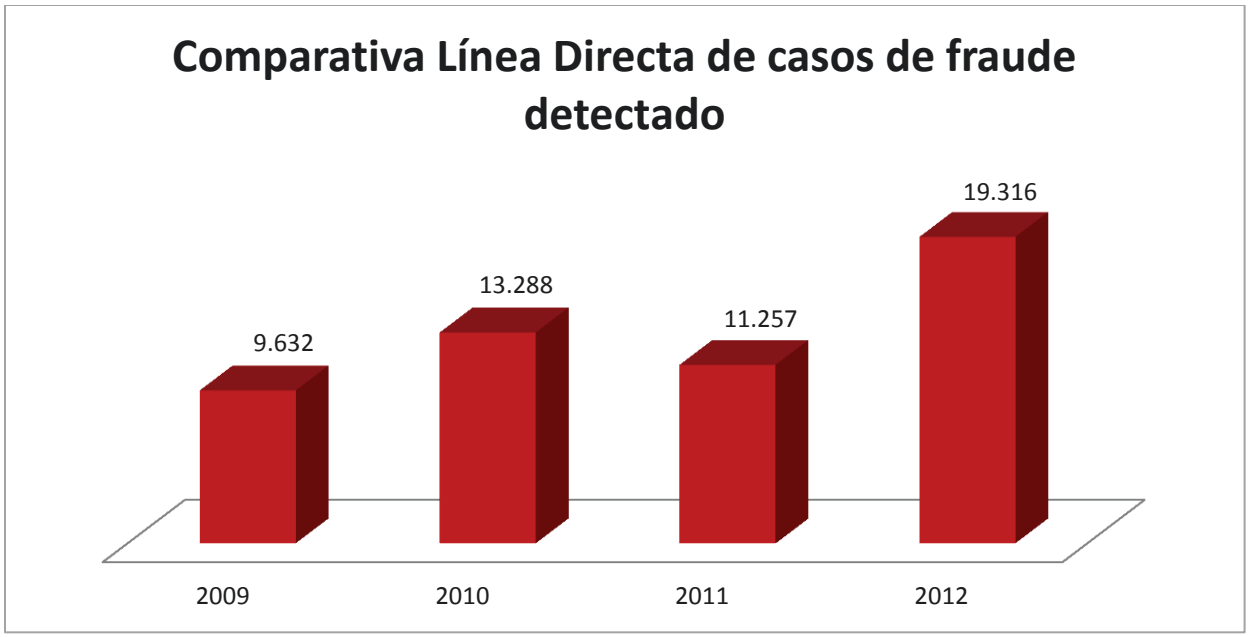
Durante mucho tiempo, el fraude fue considerado algo inevitable, un factor ineludible del riesgo, lo que contribuía a que estuviera muy extendido y que se viera como "normal". La solución que daban las aseguradoras para paliar sus efectos consistía en aumentar las primas, trasladando el problema al resto de asegurados. Uno de los principales desencadenantes de que esto fuera así radicaba en que las empresas aseguradoras consideraban la estafa al seguro como un riesgo secundario, tenían otros motivos de preocupación más grandes: la competencia en el establecimiento de primas, la captación de clientes, la aparición de nuevos riesgos o las elevadas indemnizaciones en caso de perder un juicio [10].

En los años previos a la crisis, sin embargo, la tendencia derivó hacia una mayor concienciación por parte de las compañías aseguradoras, que no podían hacer frente a las pérdidas ocasionadas por los fraudes únicamente aumentando las primas, necesitaban establecer un control y una gestión más exhaustivo de los riesgos. Comprendieron además que, a la larga, no poner medidas contra el fraude llevaría, de una u otra manera, a un aumento de éste. A raíz de esto, entre 1995 y 1996 por primera vez se establece una colaboración entre las diferentes entidades aseguradoras de automóviles, aportando datos para el concurso sectorial organizado por la asociación ICEA (Investigación Cooperativa entre Entidades Aseguradoras y Fondos de Pensiones) [11]. Esta considerado el primer estudio de tipología del fraude en nuestro país sobre el sector de automoción.

2.2.1. El fraude en España en la actualidad

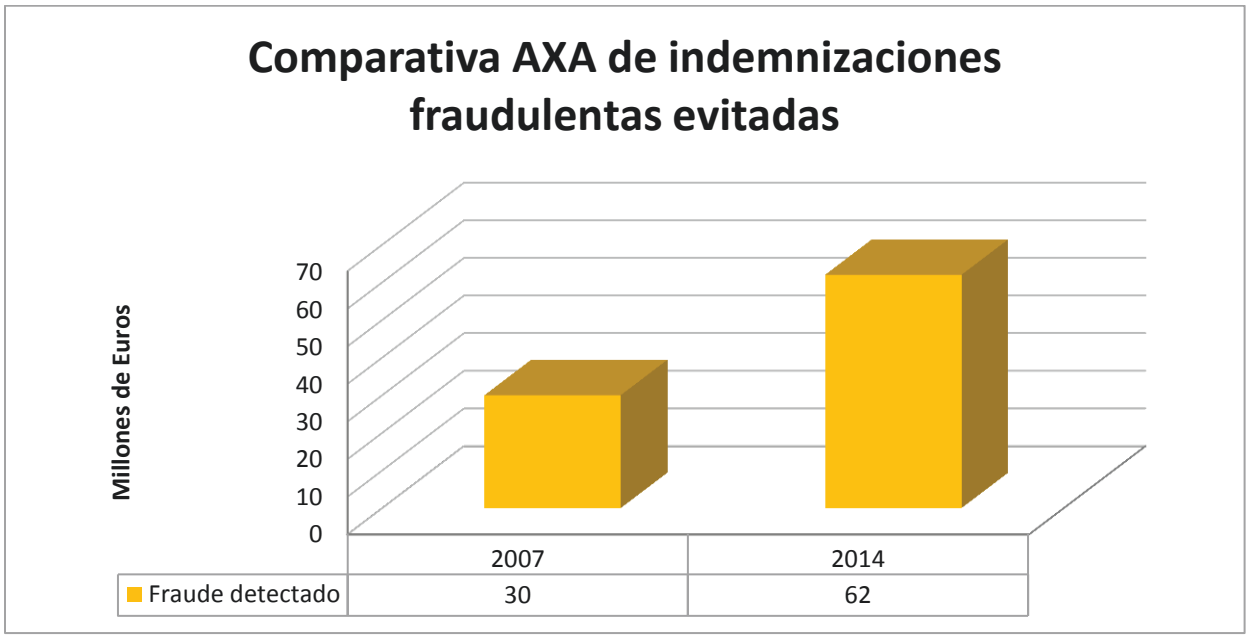
Pese a que en la actualidad cada vez existen más mecanismos y procedimientos al alcance de las aseguradoras para la detección de fraude, los últimos años no han sido especialmente buenos para el sector, presentando unas estadísticas de lo más desalentadoras.

Según el barómetro publicado en 2013 [12] por una de las empresas líderes del sector del seguro automovilístico, Línea Directa, la crisis ha disparado el fraude de una manera notable, como se puede apreciar en la Gráfica 1 de casos de fraude entre 2009 y 2012.

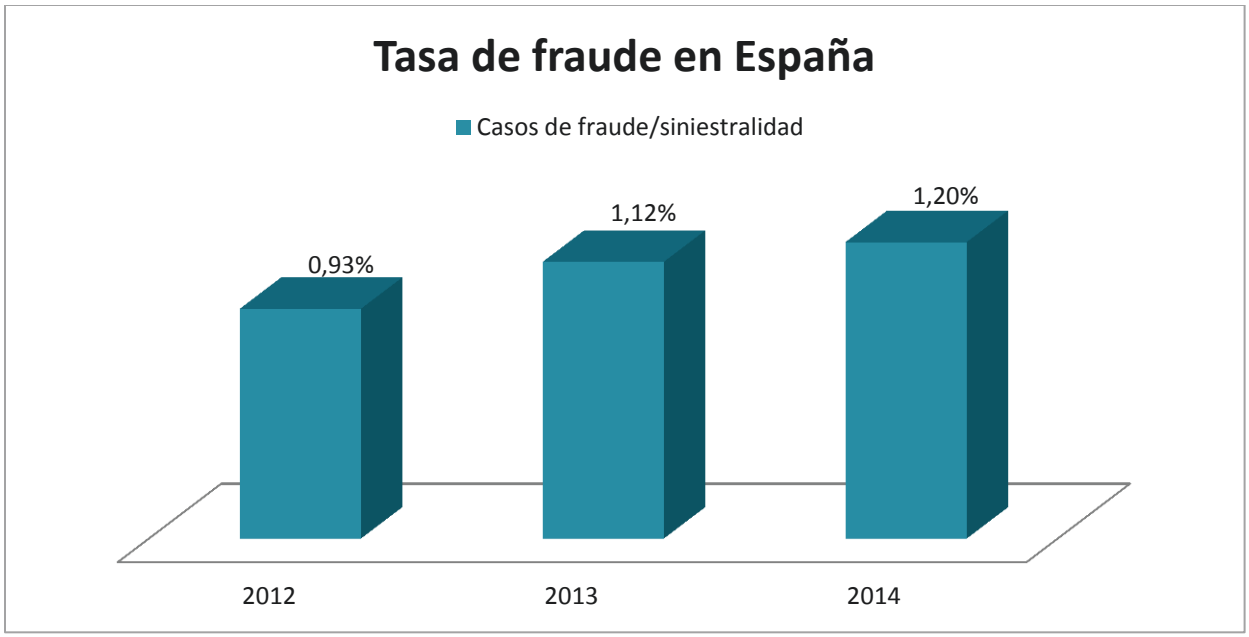


Gráfica 1: Comparativa Línea Directa de casos de fraude detectados entre 2009 y 2012

Otra de las empresas más importantes del sector, AXA Seguros, refleja en su informe de Marzo de 2015 [13] un aumento de la detección de fraude en su empresa de más del 200% entre 2007 y 2014 (ver Gráfica 2), justo desde el comienzo de la crisis hasta ahora. Los siniestros fraudulentos detectados en España en 2014 superaron los 15.300 (sólo en su empresa), un 8% más que el año anterior, a lo que se añade la tasa de fraude, o porcentaje de siniestros considerados fraudulentos con respecto al total, que siguió una tendencia alcista en los últimos años, como refleja la Gráfica 3.



Gráfica 2: Comparativa AXA de indemnizaciones fraudulentas evitadas



Gráfica 3: Comparativa AXA de tasa de fraude en España

Sobre estos datos se extraen dos conclusiones:

- La primera y más evidente es que ha aumentado el fraude en el sector del automóvil español debido a la crisis, de hecho, se ha multiplicado por dos el fraude al seguro de automóvil en este periodo [12].
- En contrapunto, la segunda conclusión destaca que se está realizando un mayor esfuerzo e inversión por parte de las empresas aseguradoras en la persecución de este tipo de actos, dado que a las aseguradoras les sale muy rentable. Según un estudio de la entidad Investigación Cooperativa entre Entidades Aseguradoras y Fondos de Pensiones (ICEA) [14], por cada euro invertido en la lucha contra el fraude, la empresa aseguradora se ahorra 43 euros en pagos fraudulentos.

Idealmente, la conciencia social y el marco legal deben dirigirse hacia un futuro donde pueda erradicarse esta práctica o al menos ser fuertemente sancionada, al igual que ha ocurrido con el fraude fiscal y empresarial que tan en liza está en estos días.

2.2.2. Tipología del fraude

El seguro automovilístico en nuestro país, al igual que en otras partes del mundo, como la Unión Europea o Estados Unidos, se basa principalmente en dos enfoques o coberturas básicas: los daños materiales y los daños corporales.

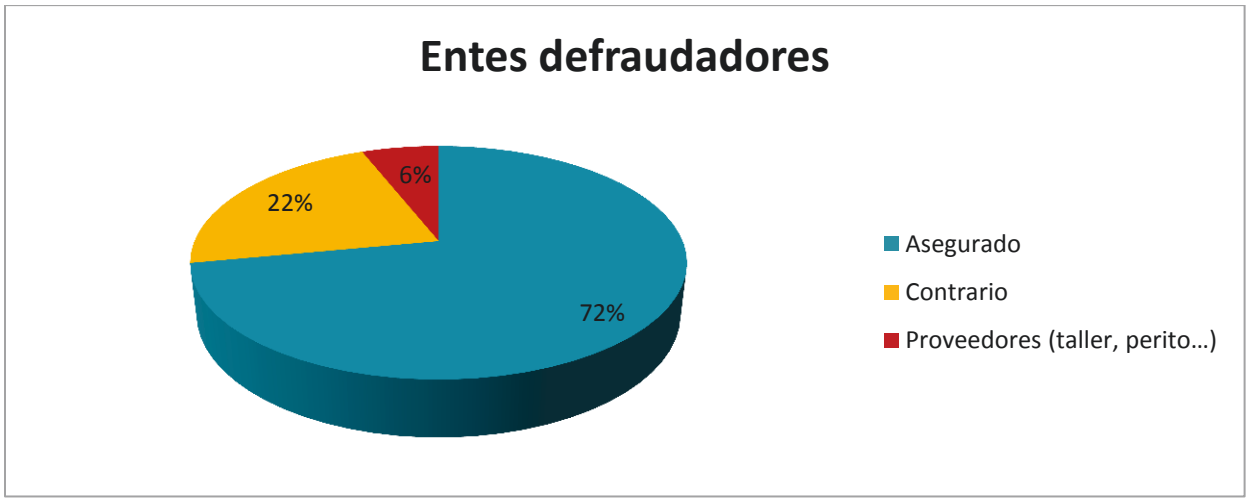
Teniendo en cuenta los casos de fraude registrados en los últimos años, aquellos referidos a daños materiales constituyen el 91% de los casos (Gráfica 4), pese a que los daños corporales son sustancialmente más costosos para las compañías aseguradoras, aunque también más difíciles de simular. Como exponen Ayuso y Santolino en su artículo de 2007 [15], para la entidad aseguradora la

compensación de los siniestros de daños corporales representa el mayor porcentaje de los costes en el seguro del automóvil. Además, son tediosos dado que los expedientes permanecen abiertos durante largos periodos de tiempo antes de ser liquidados, por lo que realizar una correcta valoración de los mismos es fundamental para la compañía.

Gráfica 4: El fraude repartido por cobertura básica

Sin embargo, debido a la naturaleza del estudio y datos de los que se dispone, este proyecto está orientado exclusivamente a los daños materiales, por lo que las tipologías detalladas a continuación se engloban dentro de dicho ámbito.

Como se expone en el estudio de Ayuso en 1995 [10], los estamentos con capacidad de fraude en el seguro del automóvil son los asegurados (incluyendo personas individuales y grupos organizados), los peritos y los talleres. Según el ICEA [14], los fraudes son cometidos en su amplia mayoría por los propios asegurados o los contrarios (otro asegurado implicado en el accidente), con un pequeño porcentaje atribuido solamente a talleres y peritos (Gráfica 5).



Gráfica 5: Comparativa ICEA de entes defraudadores

Según postula Iturgoyen (1996) [11], existe un denominador común en todos los sucesos de fraude, ya sean consumados o meras tentativas, y es el falseamiento y ocultación de datos o circunstancias en la declaración o tramitación del accidente, con la finalidad de que el asegurado o un tercero obtenga una indemnización que, de otro modo, no le correspondería. Además, el fraude cometido en España presenta la característica de ser generalmente perpetrado de manera individualizada, mientras que en el resto de Europa es más común el fraude cometido por grupos organizados.

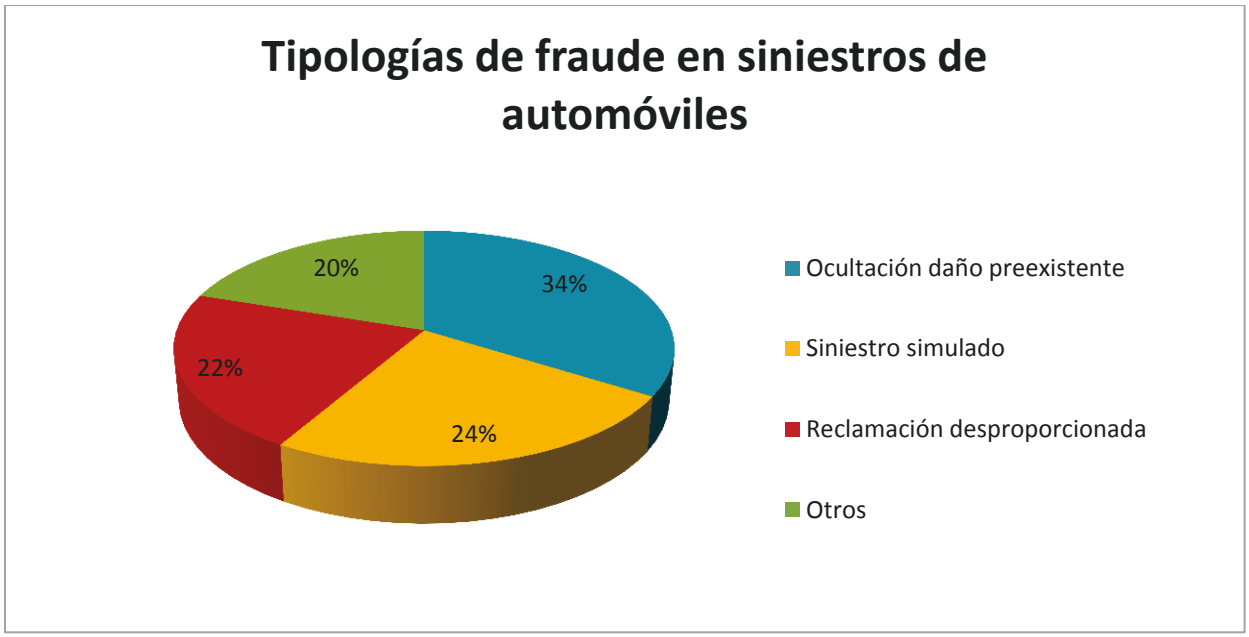
Atendiendo al fraude cometido por los asegurados o clientes, sí que pueden establecerse diferentes tipologías, contrastadas a través de numerosos estudios e informes técnicos entre los que se incluyen el del propio Iturgoyen, el de Ayuso o la publicación periódica del ICEA. A continuación se especifican los tipos más frecuentes:

- Falsa declaración del siniestro: Supone la forma más habitual de fraude, en esta categoría se suelen englobar todas aquellas acciones relacionadas con la obtención de un beneficio ilícito el asegurado, un mutuo acuerdo entre asegurado y contrario, o favorecimiento de terceros de manera deliberada. También es frecuente la ocultación de datos del siniestro no cubiertos por la póliza, la provocación de daños al vehículo por el propio asegurado de manera intencionada o la acumulación de diversos percances en un solo parte, ocultando averías o siniestros anteriores.
- Contratación de la póliza después de ocurrido el accidente: En esta circunstancia, el asegurado intenta beneficiarse de un seguro que no poseía en el momento del accidente o de unas coberturas que en ese momento no se aplicaban en la póliza. Es común que, además, este hecho también se esté produciendo con la complicidad de uno de los propios trabajadores de la empresa aseguradora.
- Falsa declaración en la póliza: Se refiere en este caso a la utilización de datos falsos en la contratación de la póliza, como los referidos a datos personales del asegurado, edad y experiencia del conductor, falso conductor habitual, antigüedad del vehículo, etcétera.
- Robo: El asegurado intenta simular un robo en su vehículo para encubrir desperfectos previos en su automóvil o para cobrar una indemnización por la desaparición del automóvil.
- Aseguramiento múltiple: En este caso, el cliente contrata diferentes pólizas para cubrir el mismo riesgo en compañías no relacionadas entre sí, para posteriormente provocar él mismo el incidente y cobrar la indemnización de todas ellas.

Trasladando el ámbito a peritos y talleres, el fraude más común en estos casos es el siguiente:

- Sobrevaloración de la tasación: Típicamente consiste en la falsificación por parte del perito de los daños acarreados por el siniestro o la elaboración de un presupuesto desmesurado por parte del taller, buscando obtener un beneficio económico. Puede hacerse en connivencia con el asegurado o no.

El ICEA realiza la siguiente comparativa de porcentajes correspondiente a 2013 [16] entre diferentes tipos de fraude, estando la mayoría contenidos en la clasificación anteriormente expuesta (ver Gráfica 6).



Gráfica 6: Comparativa ICEA de tipologías de fraude en el seguro de automóviles

Atendiendo a los datos de las gráficas anteriores y los estudios citados, puede afirmarse que la gran mayoría de intentos de fraude corresponden al sector de los asegurados, que suelen optar por ocultar información en los sucesos o simular percances para cobrar una indemnización, mientras que el fraude de peritos y talleres suele limitarse exclusivamente a la falsificación de cuantías de reparación y presupuestos.

Es bastante complicado determinar cada una de las circunstancias que rodean un siniestro y cuáles de éstas pueden constituir indicios de fraude, sin embargo, las compañías aseguradoras siguen determinadas pautas a la hora de investigar los accidentes, buscando pistas que les conduzcan a determinar si un siniestro es fraudulento o no. A continuación se listan las circunstancias más llamativas a la hora de realizar una investigación sobre un siniestro:

- Peritación: Según MAPFRE [11], el 61% de los siniestros constitutivos de fraude son detectados a través de la labor pericial o a través de ésta en combinación con otras vías. Los indicadores más frecuentes de fraude suelen consistir en que los daños no se corresponden con la mecánica del accidente, existen características que contradicen el informe (direcciones, zonas afectadas), hay restos de pintura u otras sustancias no acordes al siniestro o que la antigüedad de los desperfectos es anterior a la fecha indicada.
- Relato del accidente: Otra de las causas más importantes, en el 52% de los casos de fraude la declaración por parte del asegurado es dudosa. Además, en un 9% de estos, los declarantes se muestran nerviosos, alterados o incurren en contradicciones.
- Daños desproporcionados en el vehículo contrario: En un 25% de los fraudes descubiertos, esta circunstancia está presente. Normalmente es un indicador de que se quiere favorecer a la parte contraria.

2.3.Minería de datos

Se denomina Minería de Datos (*Data Mining* en inglés [17]) a la ciencia de explorar grandes volúmenes de datos con el fin de descubrir información implícita anteriormente desconocida y potencialmente útil. A través de un proceso computacional, se buscan patrones de conocimiento dentro de grandes conjuntos de datos, permitiendo la extracción de información de éstos y su posterior utilización para la obtención de resultados cuantificables.

Witten y Frank (2005) [18] describen la minería de datos como:

> *"El proceso de descubrimiento de patrones dentro de datos. Dicho proceso debe ser automático o (más habitualmente) semiautomático. Los patrones descubiertos deben tener sentido en cuanto a que lleven a conseguir algún tipo de ventaja, siendo ésta típicamente económica."*

Expresándolo de una manera más coloquial, el *data mining* podría compararse a la expresión "encontrar una aguja en un pajar", pero utilizando un detector de metales para acelerar la búsqueda y automatizar el proceso.

También es conveniente matizar, con el fin de obtener una mejor comprensión global del concepto en sí, que es *data mining* y que no lo es [19]. Por ejemplo, un médico que consulta una base de datos para encontrar y analizar el historial de un paciente no es *data mining*. Sin embargo, que investigadores médicos consigan agrupar a diferentes pacientes con una misma enfermedad basándose en síntomas similares, sí que lo es.

No hace más de 25 años que el concepto del *data mining* y la extracción de datos viene emergiendo en diferentes sectores y entornos, desde el campo académico a la industria, pasando por la medicina y la economía. Además, esta ciencia aún no se considera claramente atribuida a un campo de estudio en general, como postula Daryl Pregibon en 1996 [20]:

> *"El data mining es una mezcla de Estadística, Inteligencia Artificial e Investigación en Bases de Datos."*

En ocasiones también conocida como Descubrimiento de Conocimiento en Bases de Datos (*Knowledge-discovery in databases*, KDD [17]), sus orígenes se fundamentan en tres campos claramente diferenciados, o raíces principales, de las cuales ha adoptado terminología, conceptos y técnicas:

- Estadística: Se considera el pilar sobre el que se sustenta el concepto del *data mining*, así como su raíz más antigua. La estadística clásica aglutina una serie de métodos y técnicas bien definidos bajo el término comúnmente conocido como Análisis Exploratorio de Datos (*Exploratory Data Analysis*, EDA), definido por Tukey en 1961 [21]:

  > *"Componen el EDA aquellos procedimientos de análisis de datos, técnicas para la interpretación de los resultados de dichos procedimientos, formas de planificación de recolección de datos para hacer su análisis más fácil, más preciso o más exacto, y toda la maquinaria y resultados de la estadística (matemática) aplicada al análisis de datos."*

Entre las todas las técnicas utilizadas en el *data mining*, se pueden destacar las pertenecientes a la estadística descriptiva, como las distribuciones, los parámetros estadísticos tradicionales (media, mediana, desviación típica), correlaciones, análisis multivariable (*clustering*, análisis canónico, árboles de clasificación), modelos y regresiones lineales/no lineales, etcétera.

El EDA también expone técnicas para la visualización de datos, potentes métodos para la representación de grandes cantidades de datos tales como: histogramas, diagramas de caja, de dispersión, matrices, diagramas de tallo-hoja, diagramas de Pareto y otros.

- Inteligencia Artificial: De una disciplina tan vasta como la AI (*Artificial Intelligence*), el *data mining* toma prestadas técnicas de búsqueda, optimización matemática y computación evolutiva. Además, los modelos computacionales basados en el razonamiento humano son también extrapolables al ámbito del *data mining*, aportando nuevas técnicas para la extracción del conocimiento.

  La AI plantea soluciones a problemas de razonamiento humano haciendo uso de técnicas de búsqueda y optimización a través de heurísticas. La inclusión de las mencionadas heurísticas es muy necesaria si se tiene en cuenta la naturaleza compleja de los problemas del mundo real. Dado que muchos de estos problemas cuentan con espacios de búsqueda muy extensos, una mera búsqueda no informada puede requerir el uso de cantidades astronómicas de tiempo y recursos.

  Estrechamente relacionado con la AI, el Aprendizaje Automático (abordado con mayor detalle en el capítulo 2.4), representa una importante disciplina en el desarrollo del *data mining*, brindando métodos que permiten a la computadora aprender con “entrenamiento”.

- Sistemas de Bases de Datos: Considerados como el tercer pilar de la Minería de Datos, constituyen la fuente principal de información cruda para el procesamiento. Una base de datos se define formalmente [22] como:

  > *“Conjunto de datos organizado de tal modo que permita obtener con rapidez diversos tipos de información.”*

  Adicionalmente, proporcionan la organización y estructuras de datos necesarias para el tratamiento y manejo de dicha información.

El proceso de minería de datos se compone de diversas fases claramente diferenciadas, expuestas a continuación de manera pormenorizada (Ilustración 9):

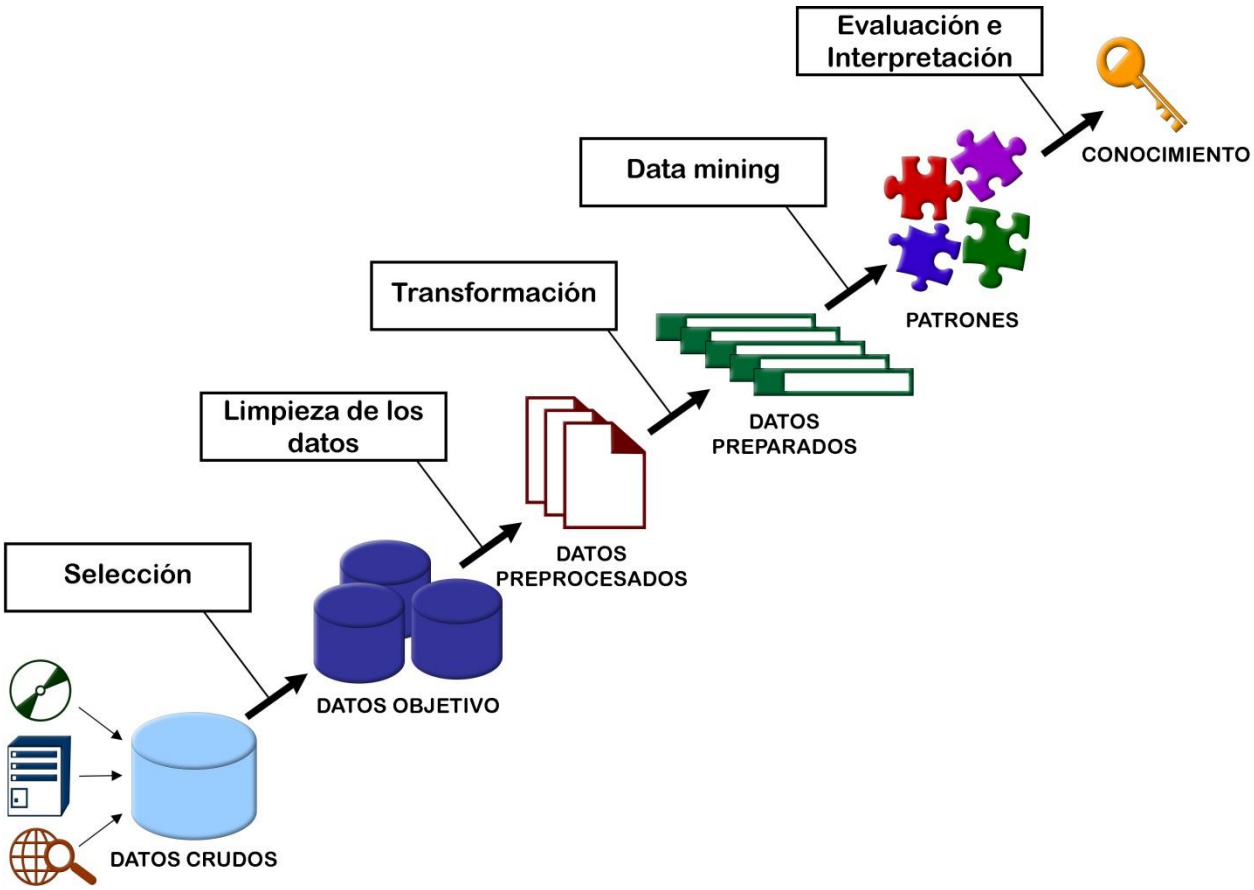


Ilustración 9: El proceso de *Data Mining*

La elaboración de un modelo de conocimiento conlleva típicamente 6 fases o pasos, que se enumeran de la siguiente manera:

- Selección de los datos: Partiendo de un muestreo general de los datos a tratar, se identificarán variables de inferencia (objetivo), variables de cálculo (independientes) y registros disponibles. Tras este proceso se obtienen uno o más conjuntos de datos que constituyen los Datos Objetivo.
- Limpieza de los datos: El siguiente paso implica un análisis previo que puede revelar información útil *a priori*, realizándose a continuación el preprocesado de los datos que típicamente consiste en la limpieza de datos no clasificados, erróneos o superfluos y eliminación de ruido.
- Transformación de los datos: Después del preprocesado, se realiza un nuevo tratamiento de los datos para dejarlos preparados para el proceso de extracción. Esta transformación es importante dado que la calidad del modelo final depende en gran medida de la calidad de los datos de entrada. Esta preparación incluye operaciones de normalización, aleatorización, reducción de dimensionalidad y separación de conjuntos para su posterior evaluación (*training-test*, validación cruzada).
- Extracción del conocimiento: Es en esta fase donde se realiza el proceso del *data mining* e identificación de patrones en sí, aplicando una o más técnicas compatibles con el tipo de los

datos a tratar (clasificación, regresión, agrupamiento, asociación, etc.). Los resultados de dichos métodos se disponen para ser analizados en la siguiente fase.

- Evaluación e interpretación: Los resultados del proceso de extracción se evalúan por medio de diferentes herramientas, elaborando como conclusión un modelo final que debe ser acorde a los resultados esperados. El proceso global puede repetirse cuantas veces sea necesario para obtener los resultados deseados o mejorar la precisión de los mismos.

Por último, cabe destacar la diferencia entre dos tipos de objetivo dentro de los problemas resolubles mediante el *data mining* [19]:

- Objetivos predictivos: Estrechamente ligados con el aprendizaje supervisado, se cumplen haciendo uso de la información contenida en las variables conocidas dentro de los datos para obtener los valores de otras variables que, de momento, son desconocidas. Dentro de esta categoría se agrupan los problemas de clasificación, regresión, predicción o análisis de series temporales.
- Objetivos descriptivos: Atribuidos al aprendizaje no supervisado, se completan mediante la obtención de patrones que definen el conjunto de datos y revelan información sobre su naturaleza. Dentro de este grupo se incluyen los problemas de *clustering, r*eglas de asociación y descubrimiento de patrones de secuencia.

2.4.Aprendizaje automático

El Aprendizaje Automático (*Machine Learning* [17]) es una disciplina englobada dentro de la Inteligencia Artificial que explora la construcción y el estudio de algoritmos que son capaces de aprender a partir de una determinada experiencia, generalmente atribuida al análisis de un conjunto de datos, y realizar una tarea de una manera más eficiente cada vez gracias a dicho aprendizaje.

El objetivo del aprendizaje automático no sólo consiste en mejorar el rendimiento de la tarea asignada, sino también dar respuesta a nuevas situaciones no conocidas utilizando la experiencia previa. La creación de un modelo computacional (hipótesis) mediante el proceso de "entrenamiento" permite realizar predicciones o tomar decisiones en situaciones no experimentadas previamente.

Según describe la Enciclopedia Británica [23], el término Aprendizaje Automático corresponde a:

> *"La disciplina que concierne al software informático que puede aprender de una manera autónoma. Los Sistemas Expertos y los programas de Data Mining son los ámbitos más habituales en el uso del Aprendizaje Automático como herramienta para la mejora de algoritmos. Entre las aproximaciones más comunes se encuentran el uso de Redes de Neuronas (caminos de decisión ponderados) y los algoritmos genéticos (símbolos criados y desechados por algoritmos para producir programas sucesivamente mejores)."*

El aprendizaje automático surgió durante los estadios iniciales de la cruzada científica en busca de inteligencia artificial. Algunos científicos se interesaron en aquellos primeros tiempos en el concepto del aprendizaje a partir de datos por parte de las máquinas, utilizando tanto modelos lógicos puramente teóricos como las recién acuñadas "redes neuronales", que imitaban la arquitectura biológica de las neuronas humanas por medio de métodos matemáticos y estadísticos.

Eventualmente ambas disciplinas han ido separándose progresivamente, tomando el aprendizaje automático una aproximación cada vez más alejada de la búsqueda de AI y enfocándose hacia la resolución práctica de problemas mediante métodos y modelos propios de la estadística, el cálculo probabilístico y la teoría de la computación [24]. También ha supuesto una gran revolución para dicha disciplina la actual disponibilidad de ingentes cantidades de información digitalizada a través de Internet.

En la actualidad se consideran diversos tipos de problemas/tareas de aprendizaje automático: Aprendizaje Supervisado, Aprendizaje No Supervisado y Aprendizaje Por Refuerzo [25]. En este proyecto se hace especial hincapié en los dos primeros.

2.4.1. Aprendizaje Supervisado

Se define como la tarea de aprendizaje automático consistente en inferir un modelo a partir de un conjunto de datos de entrenamiento de los que se conoce el resultado, para posteriormente aplicar dicho modelo a nuevos datos, de los que no se conoce la solución *a priori*, para que sean clasificados en base a ese patrón previamente planteado.

El conjunto de entrenamiento consiste típicamente en un muestreo de ejemplos de entrenamiento compuestos por una serie de datos de entrada y una salida deseada. Un algoritmo de aprendizaje supervisado analiza dichos ejemplos y produce una función o modelo que será utilizada para predecir la salida en nuevos ejemplos.

A continuación, se analiza un segundo conjunto, llamado conjunto de *test*, que prueba el modelo anteriormente establecido pero esta vez sin contar con salidas deseadas. Un escenario ideal sería aquel en el cual el algoritmo fuera capaz de generalizar las nuevas instancias nunca vistas de una manera "razonable".

Llegado este punto, un buen modelo será capaz de clasificar el conjunto de *test* con precisión sin caer en el denominado "pecado capital del *data mining*" [26], el *overfitting* o sobreajuste a los datos. El *overfitting* se manifiesta cuando un modelo es preciso sobre el conjunto de entrenamiento, consiguiendo buenos resultados, pero sobre el conjunto de *test* no consigue generalizar de una manera adecuada y se dispara el error cometido en la clasificación (Ilustración 10). Se dice entonces que está sobreajustado a los datos de entrenamiento.

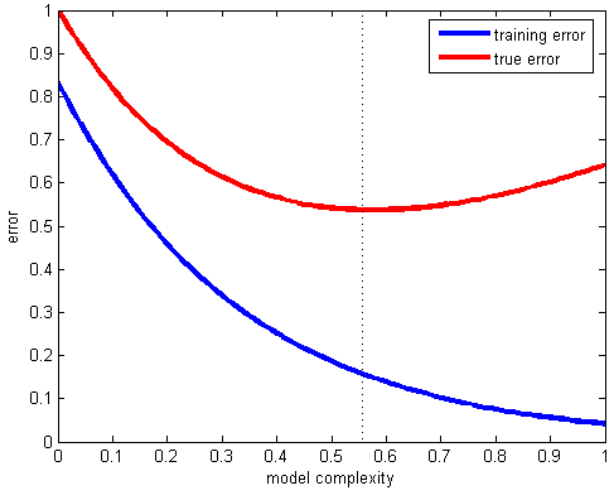


Ilustración 10: Ejemplo de *overfitting*

Los datos aportados para el presente proyecto presentan una salida deseada en forma de atributo o etiqueta de clase (la descripción de los datos está disponible en la sección 3.4), por lo que lo más razonable será utilizar algoritmos diseñados para problemas de clasificación, denominados clasificadores. Entre los clasificadores más comunes se encuentran los árboles y reglas de decisión, las redes neuronales, los clasificadores bayesianos y las máquinas de vectores de soporte (SVM).

Durante la experimentación del proyecto se han utilizado los siguientes clasificadores:

- Árboles de decisión: Considerada una de las alternativas más populares y elegantes a la hora de trabajar con problemas de clasificación, en ella el modelo se representa en forma de árbol, donde se establecen una serie de acciones posibles de realizar en función de una o varias

variables. Atendiendo a los valores que tomen estas variables, el árbol se va bifurcando desde los nodos a las hojas, trazando diferentes caminos que terminan en una acción determinada (nodo hoja). Para el caso particular de un problema de clasificación, los diferentes valores que toman las variables de entrada (atributos) marcarán un camino que desembocará en la decisión de clasificar dicha instancia como de una clase o de otra (Ilustración 11).

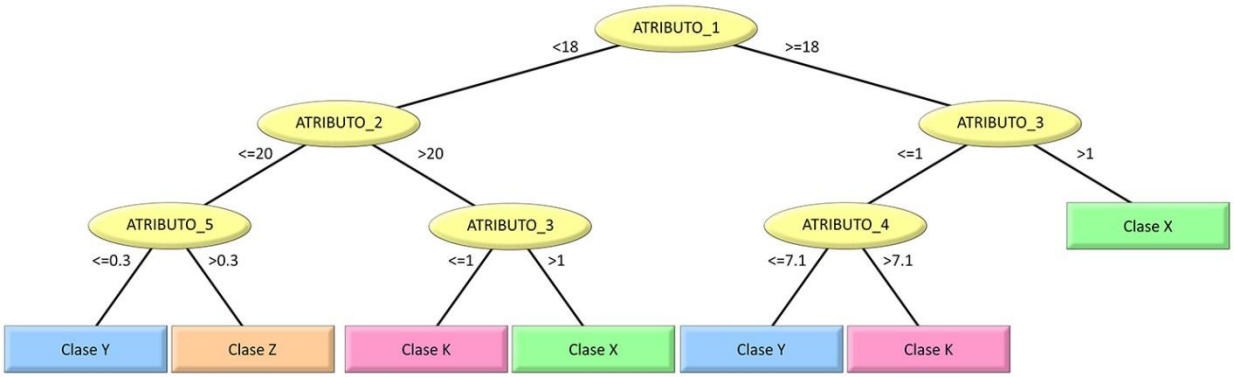


Ilustración 11: Ejemplo de árbol de decisión

Como implementación del clasificador de árboles de decisión, se ha decidido utilizar para el análisis el algoritmo C4.5, desarrollado por John Ross Quinlan [27]. Se trata de una expansión del algoritmo ID3, también desarrollado por Quinlan en 1983. La clave de ambos algoritmos radica en la elección de atributos para la subdivisión del árbol mediante la minimización de la entropía de información (o maximización de la ganancia de información) [28]. El atributo con mayor ganancia de información en cada iteración del algoritmo se selecciona como parámetro de decisión. Esto significa que aquellos atributos con mayor ganancia de todo el conjunto irán sucesivamente estableciéndose en el árbol como nodos desde la raíz a las hojas. Además, la introducción de la post-poda en este algoritmo permite limitar el sobreajuste a los datos de entrenamiento y mejora la generalización del modelo. La aplicación de minería de datos WEKA (Capítulo 3.1) ofrece una implementación *open-source* del algoritmo C4.5 programada en lenguaje *Java* denominada J48.

- Reglas de decisión: Las reglas de decisión suelen ser utilizadas, en combinación con árboles, para poder aplicar técnicas como la poda, útil para paliar los efectos del sobreajuste del modelo a los datos de entrenamiento. Los árboles de decisión producidos por C4.5 pueden ser traducidos a reglas de decisión, más fácilmente interpretables a la hora de realizar la post-poda de reglas sobrantes. Sin embargo, algunos científicos mencionan la poca efectividad de la poda en muchos problemas del mundo real, y abogan por la utilización de métodos más simples, también basados en reglas de asociación, como el método de las "*1-rules*" (o 1R). Desarrollado por Robert C. Holte en 1993 [29], 1R consiste en crear diferentes reglas de clasificación basadas únicamente en un único atributo. Dado que el conjunto de entrada engloba diferentes variables o atributos, el método prueba con todos ellos y elije el modelo que devuelva el error mínimo de clasificación. Pese a parecer tosco y simple, en la gran mayoría de los casos, sorprende con

modelos de clasificación bastante precisos. WEKA ofrece una implementación *open-source* de este método denominada OneR.

2.4.2. Aprendizaje No Supervisado

En esta variante del aprendizaje automático, el algoritmo intenta encontrar una estructura implícita dentro del conjunto de datos sin etiquetar, es decir, no se cuenta con entrenamiento previo. Al carecer de conjunto de entrenamiento, el tratamiento de los datos se hace directamente sobre todo el conjunto de entrada, generando un modelo basado en un análisis de densidad de los datos que componen el conjunto total. Ésta es la principal característica que distingue el aprendizaje no supervisado del supervisado o del aprendizaje por refuerzo.

Existen diferentes aproximaciones para el aprendizaje no supervisado, siendo las siguientes las más destacadas:

- *Clustering*: Este método radica en agrupar diferentes objetos de tal manera que aquellos que pertenecen a un grupo (llamado clúster) se asemejen más entre sí que a aquellos presentes en otros grupos. Para determinar la pertenencia de un elemento a un grupo, suelen tenerse en cuenta criterios como la distancia a otros ejemplos o la similitud en otros términos. El *clustering* en sí no se trata de un algoritmo, sino de una tarea que puede ser alcanzada con diferentes algoritmos.
- Modelos de aprendizaje de variables latentes: Consiste en la inferencia mediante un modelo matemático de variables "ocultas" o latentes en el conjunto de datos a través de otras variables que si son observables.

Existen multitud de algoritmos y técnicas asociadas con el aprendizaje no supervisado, como el algoritmo de K-medias, el de Esperanza-Maximización (EM), Cobweb o, relacionados con las redes neuronales, los mapas auto-organizados (SOM) y la Teoría de la Resonancia Adaptativa (ART). En este proyecto en particular se utilizarán exclusivamente dos algoritmos de *clustering*¸ K-medias y EM.

- K-medias (*K-means*): Pese a ser acuñado por MacQueen en 1967 [30], originalmente su planteamiento se atribuye a Stuart Lloyd en 1957, por lo que también es conocido como "Algoritmo de Lloyd". Considerado uno de los métodos más populares de agrupación de datos debido a su simplicidad, el algoritmo consiste en el particionamiento del conjunto de datos inicial en diferentes subconjuntos (o clústeres), creando zonas diferenciadas en el espacio de datos, denominadas comúnmente polígonos de Thiessen o regiones de Voronoi. El agrupamiento de los datos se realiza atendiendo a la proximidad de los mismos a determinados prototipos representativos de cada subconjunto, que van siendo recalculados en cada iteración. La medida de proximidad se establece mediante el cálculo de distancias, siendo las distancias euclídea y Manhattan las más comúnmente utilizadas.

  La mecánica del algoritmo es, a grandes rasgos, la siguiente [31]:

  1. Se inicializan *k* prototipos ($w_1,\ldots,w_k$), y a cada uno le es asignado uno de los *k* subconjuntos o clústeres, al cual representa ($C_1,\ldots,C_k$).

2. Se procesan todos los vectores de entrada, siendo cada vector asignado al clúster C_j cuyo prototipo esté más próximo (en distancia).
3. Una vez procesadas todas las instancias, se calcula el centroide de cada clúster C_j teniendo en cuenta todas las instancias contenidas en ese momento en el clúster. El centroide se convierte a continuación en el nuevo prototipo w_j.
4. Se repite desde el paso 2 hasta que se alcance un momento en el cual las asignaciones no cambien o el cálculo del error se estabilice.

En la Ilustración 12 se muestra de manera gráfica el proceso:

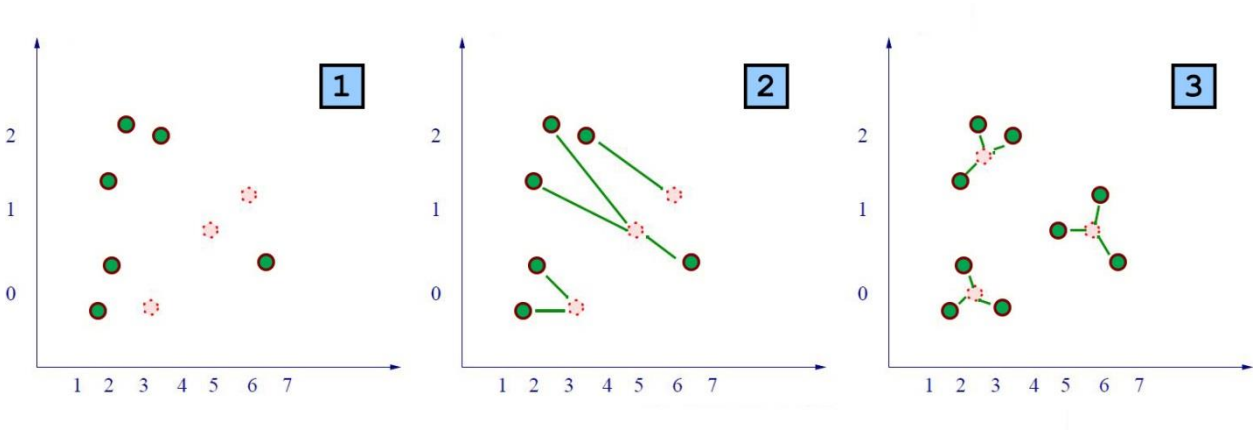


Ilustración 12: Calculo de centroides en K-medias

- EM (*Expectation-Maximization*): Propuesto y acuñado por Dempster, Laird y Rubin en 1977 [32], se trata de un algoritmo iterativo comúnmente utilizado en problemas donde los datos están o se consideran incompletos. Su uso está muy extendido en diferentes áreas, como la visión artificial, el procesamiento de voz o el reconocimiento de patrones. EM tiene por objetivo particionar los datos en diferentes clústeres de tal manera que se consiga máxima verosimilitud en los parámetros de cada uno de esos clústeres [31]. El algoritmo itera alternando entre un paso de esperanza (*E-step*), donde se calcula la verosimilitud esperada a través de la inclusión de variables latentes, y un paso de maximización (*M-step*), donde se maximiza la verosimilitud establecida en el paso E a través del cálculo de estimadores de máxima verosimilitud (los mencionados parámetros de cada clúster). Por último, estos parámetros se usan para el siguiente *E-step* (Ilustración 13).

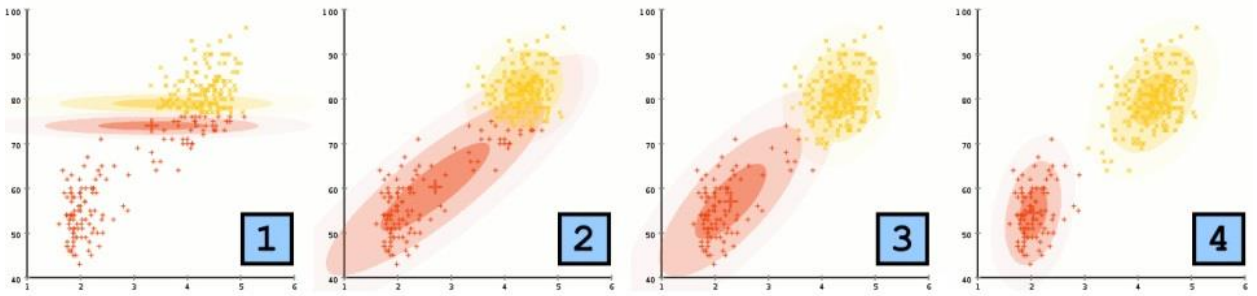


Ilustración 13: Cálculo de clústeres mediante EM

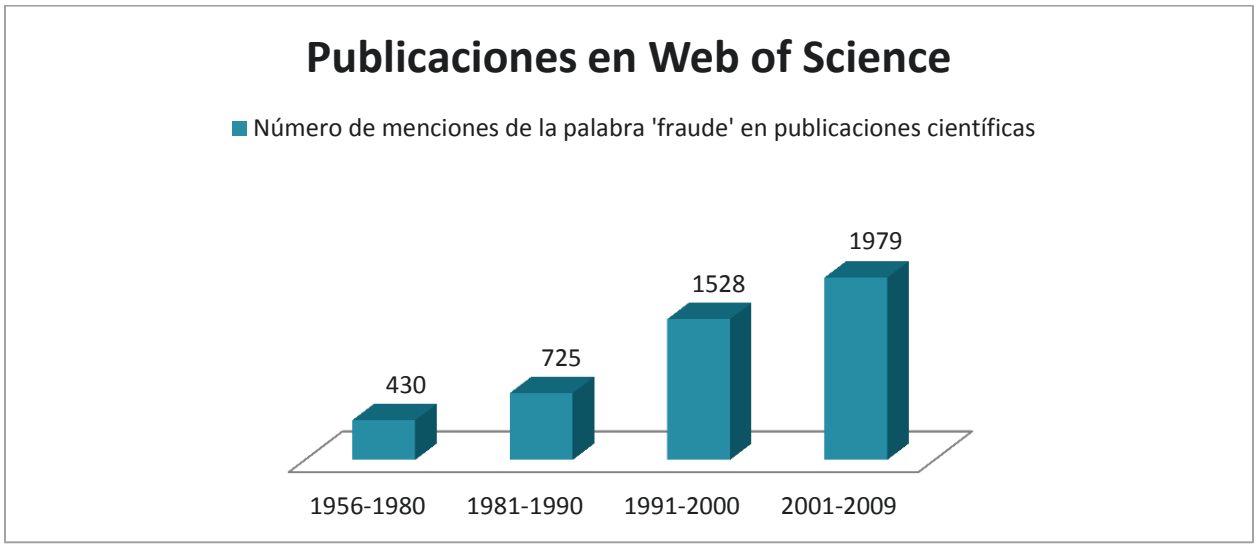
2.5.Estudios previos y proyectos similares

A día de hoy se pueden encontrar numerosos estudios estadísticos y técnicos relacionados con la detección y prevención del fraude en los seguros, lo que parece lógico teniendo en cuenta el volumen de mercado y movimiento de dinero que genera este sector. Puede afirmarse sin ningún género de duda que la detección de reclamaciones fraudulentas es una de las principales fuentes de mantenimiento de la rentabilidad en la industria aseguradora, es por ello que la inversión a la investigación sobre este ramo es bastante importante en la actualidad.

Tradicionalmente, las compañías aseguradoras en el pasado detectaban posibles casos de fraude mediante la inspección de bases de datos llevada a cabo por técnicos (inspección manual), o utilizando métodos estadísticos tradicionales y reglas heurísticas. En la actualidad, a medida que el tamaño de las bases de datos crece, estos métodos pueden perder eficacia y pasar por alto numerosos casos de fraude debido a dos motivos:

- Actualmente es imposible detectar todos los casos de fraude mediante la inspección manual de grandes bases de datos. La mera tarea de revisar todos los informes puede conllevar años de trabajo.
- Surgen nuevas formas de fraude continuamente. Los métodos heurísticos no son válidos a la hora de detectar nuevos perfiles de estafa, sino que son necesarias otras herramientas capaces de "actualizarse" para contrarrestar este fenómeno.

Está claro que la ciencia cada vez representa un papel más importante en la investigación del fraude, pero, ¿es el fraude un asunto de interés actualmente para la ciencia? Según el estudio publicado por Pejic-Bach en 2010 [33], se puede corroborar que la ciencia cada vez investiga más sobre el fenómeno del fraude. El estudio se basa en el recuento de menciones o citas en artículos científicos publicados en las bases de datos de la Web of Science desde 1956 que contengan la *keyword* "fraude". En la Gráfica 7 se puede observar la evolución hasta 2009.



Gráfica 7: Comparativa de publicaciones en Web of Science

Para la realización de este proyecto se han consultado gran cantidad de artículos, libros y otras publicaciones científicas correspondientes a los últimos años. La gran mayoría de estudios encontrados se basan en análisis y detección de fraude orientado a reclamaciones de daños corporales y lesiones derivadas de los accidentes de tráfico, que pese a no ser exactamente el ámbito que se quiere tratar, brindan soluciones y conclusiones útiles para el objetivo de este proyecto.

Entre los estudios más destacados figura el realizado por Derrig y Ostaszewski (1995) [34] acerca del reconocimiento de patrones mediante técnicas de lógica difusa y *clustering* (sección 2.4.2) en la clasificación de riesgos y reclamaciones de daños personales en pólizas pertenecientes al estado de Bristol, Massachusetts, en Estados Unidos. En la publicación se describe la técnica utilizada como un método de agrupamiento semejante al k-medias utilizando lógica difusa, que permite agrupar en diferentes clústeres las reclamaciones con mayores puntos en común entre sí, y por tanto, aquellas susceptibles de fraude dentro de la misma categoría dado que se considerarán semejantes. Este estudio es quizá el más interesante, dado que toda la experimentación se realiza sobre un ámbito y con un tratamiento de datos muy parecido al que se desarrollará en este proyecto.

En otro artículo de la misma temática de 2004, Viaene, Derrig y Dedene [35] proponen una aproximación diferente utilizando un clasificador Bayesiano ingenuo mejorado mediante *AdaBoost*, de nuevo tratando datos de siniestros con daños corporales en Massachusetts. El objetivo es establecer una puntuación sobre cada instancia atendiendo a características individuales que sean sospechosas de conllevar fraude, y mediante un sencillo mecanismo de agregación ofrecer la puntuación antes mencionada.

Otro trabajo orientado a la lógica difusa es el publicado por Bordoni y Facchinetti (2001) [36] sobre sistemas expertos aplicados a reclamaciones de compañías aseguradoras en Italia. En el artículo se describe un "Controlador de Lógica Difusa" (*Fuzzy Logic Control*, FLC) que permite establecer un *ranking* o una puntuación asignada a cada siniestro atendiendo a ciertas variables de entrada. Aquellos que superen cierto umbral serán catalogados como sospechosos de fraude e investigados por un equipo técnico al uso.

También es importante mencionar el trabajo de Yi Peng, Gang Kou et al. (2006) [37] sobre métodos de *clustering* para la detección de fraude en seguros de salud. Este estudio se considera importante por el tratamiento que utiliza para el entendimiento de los *datasets* y herramientas de minería de datos (las utilizadas para este proyecto se detallan en la sección 3.1).

Un último artículo importante de destacar es el desarrollado por Hajian, Domingo-Ferrer y Martínez-Ballesté (2011) [38], basado esta vez en el uso de reglas de asociación para encontrar características que marquen tendencias dentro de grandes conjuntos de datos. Pese a no tener una temática relacionada (en este caso se trata de prevención de casos de discriminación en asaltos y delitos), la exposición de técnicas basada en reglas de asociación es interesante para el presente proyecto.

También se ha extraído información puntual de otros estudios como el de D'Arcy (2005) [39], Yoo et al. (1994) [40] o Tseng y Lu (2011) [41].

De los anteriores estudios se pueden sacar en claro diversas cuestiones. Las dos técnicas más comunes para detectar casos fraudulentos suelen ser:

- *Rankings* de puntuación con un umbral de investigación.
- Agrupación en clústeres atendiendo a similitud de características.

Para alcanzar estos objetivos, el uso de técnicas de *data mining* como el *clustering*, la lógica difusa o las redes neuronales son las alternativas más adecuadas para realizar el análisis.

No se ha encontrado ningún estudio exclusivamente basado en el fraude de daños materiales en el seguro del automóvil, pero se tomaran soluciones, métodos y técnicas reflejadas en los documentos anteriores.

3. Diseño de la solución

3.1.Herramientas utilizadas

Para el desarrollo de este proyecto se han utilizado diferentes herramientas y aplicaciones, que se detallan a continuación:

El desarrollo de la experimentación se ha llevado a cabo prácticamente en su totalidad con la aplicación WEKA 3.7 de la Universidad de Waikato (Nueva Zelanda), debido a su gran biblioteca de herramientas y algoritmos para el *data mining*: regresión, clasificación, clustering, reglas de asociación, visualización y tratamiento de ficheros de datos de gran tamaño [42].

Ilustración 14: Logotipo de WEKA

Ilustración 15: Logotipo de Adobe Photoshop

Se han incluido determinados *plugins* para WEKA que amplían su funcionalidad, como PrefuseTree o ScatterPlot3D.

Para la elaboración de gráficas, histogramas y otros diagramas de análisis de datos se ha utilizado Microsoft Excel 2010, que permite la creación de gráficos lineales sencillos, elegantes y vistosos.

Para el tratamiento general de los ficheros de datos que se manejan en WEKA se ha utilizado la aplicación Notepad++.

En la realización de diagramas de Gantt relacionados con la planificación, se ha utilizado la herramienta *online* gratuita Gantter.

Por último, se ha utilizado la herramienta Adobe Photoshop CS6 para la elaboración de determinados diagramas explicativos, así como para el tratamiento de las imágenes incluidas en este documento.

Debido a que se trata de un proyecto de investigación, no se ha considerado conveniente desarrollar ningún programa propio que dé solución al problema planteado, únicamente se han utilizado plataformas ya existentes y documentación. Dicho esto, sí que se han desarrollado pequeñas utilidades de apoyo en código Java para completar ciertas tareas relacionadas con la recogida de datos.

3.2.Estructura del proyecto

El proyecto que se va a acometer se define esencialmente como un trabajo de análisis de datos pormenorizado sobre un conjunto de instancias procesado de antemano. La experimentación, que se valdrá de diversas técnicas y métodos de *data mining*, se especifica a continuación.

El fichero que contiene los datos fue sometido a un proceso de *clustering* previo al proyecto, que devolvió una serie de asignaciones guardadas en el propio *dataset* estudiado. Éstos atributos de clase actúan como "salida deseada" para los algoritmos de aprendizaje supervisado, en el caso de que éstos se utilicen.

El presente trabajo se divide en dos ramas principales: el análisis de grados de severidad y la auditoría de daños. El análisis de severidad se basa en los resultados del *clustering* previo al proyecto para extender la experimentación y obtener modelos de conocimiento aplicables en la lucha contra el fraude, mientras que la auditoría de daños se centra en realizar diversas pruebas sobre el *dataset* con el fin de establecer un nuevo agrupamiento de los datos basado en tipologías de impacto.

La primera etapa del análisis de grados de severidad consiste en emular el proceso de *clustering* previo para poder obtener más información acerca del conjunto de datos y establecer las categorías por las cuales se agrupan las instancias. Se utilizarán algoritmos de aprendizaje no supervisado para dicho agrupamiento u otros algoritmos que se consideren convenientes y que puedan aportar mayor conocimiento sobre los datos. Antes de iniciar las pruebas será necesario pretratar los datos para adecuarlos a las necesidades del problema (ver sección 3.6).

En una segunda fase se amplía el estudio sobre los resultados obtenidos en el proceso de *clustering*. Mediante algoritmos de clasificación supervisada, se pretende conseguir un modelo basado en reglas de decisión que "retrate" la clasificación por grados de severidad y permita clasificar instancias futuras con la mayor precisión posible. Este modelo podrá utilizarse para contrastar valoraciones sospechosas de ser fraudulentas.

En cuanto a la auditoría de daños, se pretende conseguir un agrupamiento completamente nuevo que permita delimitar las zonas e intensidades de impacto atendiendo a las reparaciones más típicas. Para ello, se realizarán diferentes pruebas con algoritmos de *clustering* que obtengan una distribución diferente a la original. Idealmente, aquellas instancias que presenten los desperfectos más representativos en cada zona del vehículo se agruparán en *clústeres* similares, lo que permitirá en última instancia perfilar un mapa de tipologías de impacto atendiendo a grupos de piezas, ubicación, intensidad y trayectorias de los daños del siniestro.

Finalmente, se elaborarán una serie de conclusiones al respecto de las dos aproximaciones y se discutirá sobre posibles aplicaciones futuras del proyecto en la lucha contra el fraude.

En el siguiente diagrama se explican de manera gráfica las etapas definidas anteriormente:

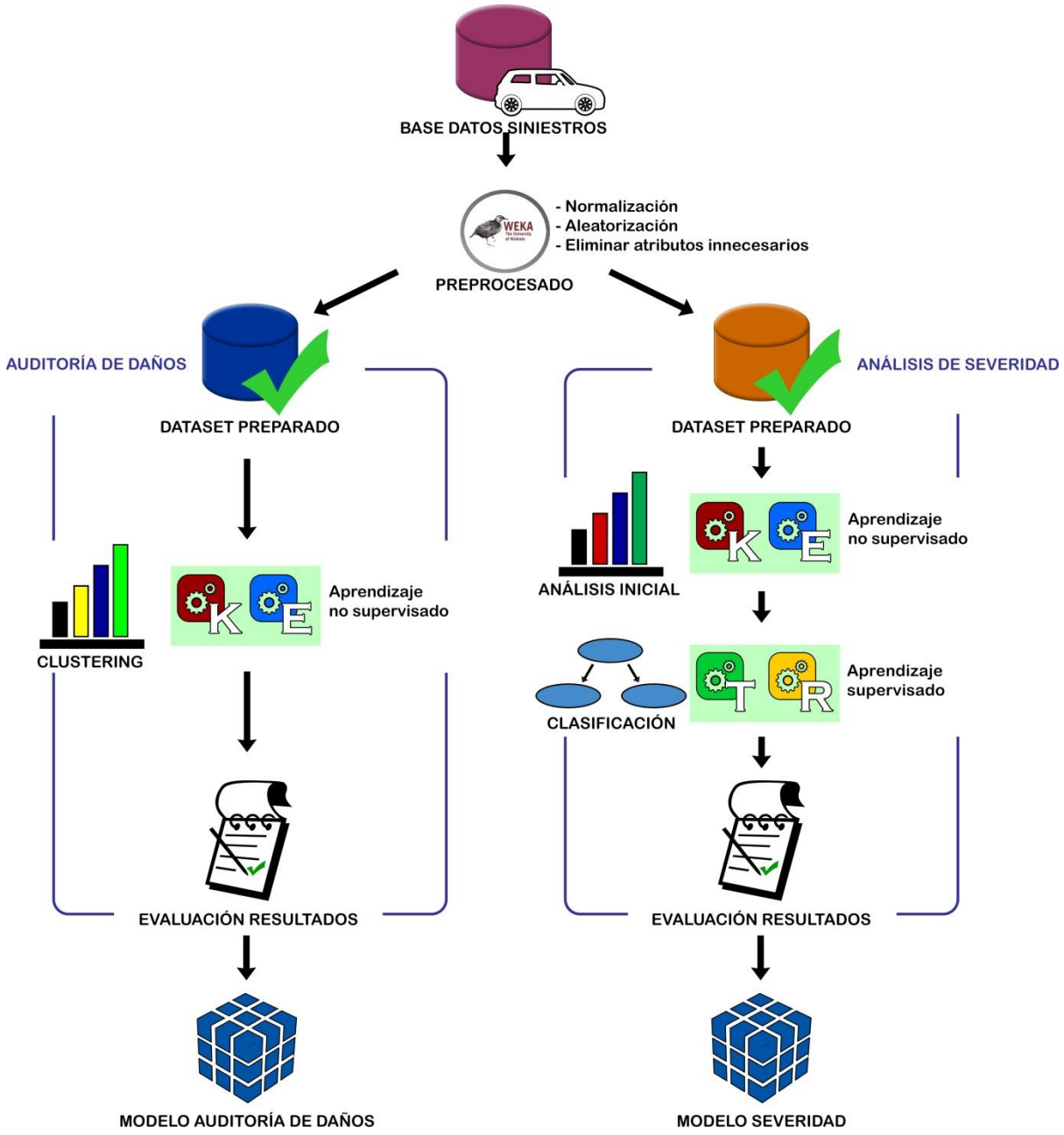


Ilustración 16: Estructura del proyecto

En la Ilustración 16 se representan a grandes rasgos las diferentes etapas del proyecto. Aquellas partes enlazadas con flechas en rojo corresponden a fases que no se han acometido en el presente proyecto y que pueden desarrollarse en el futuro.

Debido a que se carece de registros que alberguen ejemplos de siniestros fraudulentos, es necesario matizar que los modelos que se extraerán de la experimentación no permitirán clasificar nuevas instancias como fraudulentas o no, sino ejercer de mecanismo evaluador para poder contrastar dichas instancias y que sean catalogadas por terceros.

3.3.Marco regulador

Debido a la naturaleza de los datos y la pertenencia de éstos a una base de datos privada, este proyecto podría verse afectado por la legislación vinculada a la Ley Orgánica de Protección de Datos (LOPD). Sin embargo, el preprocesamiento inicial al que fue sometido en su día el *dataset* ha permitido anonimizar las instancias, eliminando todo rastro de datos personales o identidades que puedan relacionarse con terceras personas.

Por tanto, no hay ningún marco regulador que afecte al proyecto.

Sin embargo, cabe mencionar que la empresa propietaria ha exigido firmar un contrato de confidencialidad al respecto de la fuente de los datos para este estudio.

3.4.Descripción de los datos

Inicialmente se dispone de un fichero previamente procesado a partir de una Base de Datos de una importante compañía de seguros española, con datos de siniestros correspondientes a años pasados. Este fichero original cuenta con alrededor de 330.000 instancias pero, atendiendo a razones de rendimiento y carga de procesamiento, se ha decidido utilizar un muestreo reducido de 27.706 instancias para la experimentación. Se considera que este volumen es suficiente para garantizar resultados. Excepcionalmente, se podría ampliar el número de instancias si es necesario.

El archivo presenta un formato preparado para ser introducido en la aplicación WEKA (Tabla 1), en la que cada instancia está formada por una tupla o vector de 115 atributos, divididos en diversos tipos. Para una consulta detallada, la guía completa de atributos se encuentra en el ANEXO B: Desglose de atributos.

```
@relation SalidaBinario_clustered

@attribute Instance_number numeric
@attribute Secuencia numeric
@attribute Historia numeric
@attribute PT_51503 {0,1}
@attribute PT_53203 {0,1}
...
@attribute REP_51503 {0,1}
@attribute REP_53203 {0,1}
...
@attribute SUST_94102 {0,1}
@attribute SUST_94202 {0,1}
@attribute Pos_int numeric
@attribute Pos_mod numeric
@attribute Tot_mo numeric
@attribute Tot_pint numeric
@attribute Tot_piez numeric
@attribute Tot_gen numeric
@attribute Cluster {cluster0,cluster1,cluster2,cluster3}

@data
0,2003614760,1,1,0,1,1,1,1,0,1,0,1,0,1,0,1,0,0,1,0,1,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,32,2,138.7,674.91,26.98,840.59
,cluster3
1,2019289963,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,5,2,75.77,128.19,21.31,225.27
,cluster2
```

Tabla 1: Formato de fichero de datos para WEKA

A continuación se exponen con detalle cada uno de los tipos de atributo:

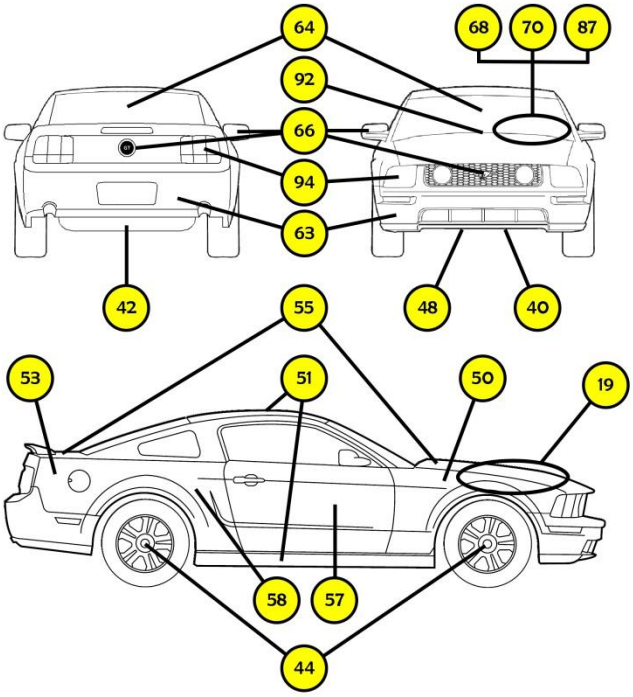
- Atributos numéricos: Existen 9 atributos numéricos que almacenan los siguientes valores de cada tupla:
    - Instance_number: Representación del número de cada instancia, con un rango de 0 a 27.705.
    - Secuencia: Clave de identificación basada en el número de referencia del siniestro, compuesta por 10 dígitos.

- Historia: Almacena el número de ocasiones en la cual el parte ha sido revisado y valorado de nuevo debido a que se necesitan reparaciones no registradas en el informe. Por defecto su valor es 1.
- Pos_int: Número de piezas sustituidas en esa instancia concreta.
- Pos_mod: Número de piezas modificadas (reparadas) en esa instancia concreta.
- Tot_mo: Coste total de la mano de obra.
- Tot_pint: Coste total del trabajo de pintura.
- Tot_piez: Coste total de las piezas sustituidas.
- Tot_gen: Coste general de todas las reparaciones, suma de los costes anteriores.

- Atributos booleanos: El fichero también incluye 105 atributos booleanos, cada uno correspondiente a un grupo de piezas específico dentro de la mecánica del coche. Debido a la ingente cantidad de piezas que contiene un vehículo, se ha decidido realizar esta agrupación de piezas similares para aligerar el procesamiento y simplificar la nomenclatura. Cada uno de estos atributos cuenta con un identificador unívoco (por ejemplo REP-51751) que establece su grupo de piezas, familia y lateralidad. En la siguiente Tabla 2, se detalla la nomenclatura utilizada para el nombrado de dichos atributos.

PT-XXYYZ	• PT: Tipo de reparación. Indica el tipo de reparación aplicado sobre el grupo de piezas: PT (pintura), REP (reparación) o SUST (sustitución). • XX: Número de grupo. Cada grupo de piezas se distingue por los primeros dos dígitos del identificador global. Consultar la Ilustración 17 para una referencia más detallada sobre los grupos de piezas. • YY: Número de familia. Dentro de cada grupo de piezas existen familias para realizar una subdivisión más precisa. Cada familia se identifica con las segundas dos cifras del identificador global. • Z: Lateralidad de la pieza. Indica el lado en aquellas piezas que suelen estar presentes en ambos lados del vehículo: 1 (izquierda), 2 (derecha) y 3 (central/sin lateralidad).

Tabla 2: Identificador de atributos booleanos



Guía de referencia para grupos de piezas:
19 - Sistema de refrigeración del motor
40 - Suspensión y eje delanteros
42 - Suspensión y eje traseros
44 - Ruedas (Neumáticos y llantas)
48 - Dirección
50 - Componentes frontales de carrocería
51 - Componentes centrales de carrocería
53 - Componentes traseros de carrocería
55 - Capó y portón trasero
57 - Puerta delantera
58 - Puerta trasera
63 - Paragolpes delantero y trasero
64 - Lunas y componentes
66 - Componentes unidos a la carrocería (espejos, rejillas, matrículas)
68 - Sistemas de seguridad
70 - Componentes internos de carrocería
87 - Aire acondicionado
92 - Limpiaparabrisas
94 - Faros y pilotos

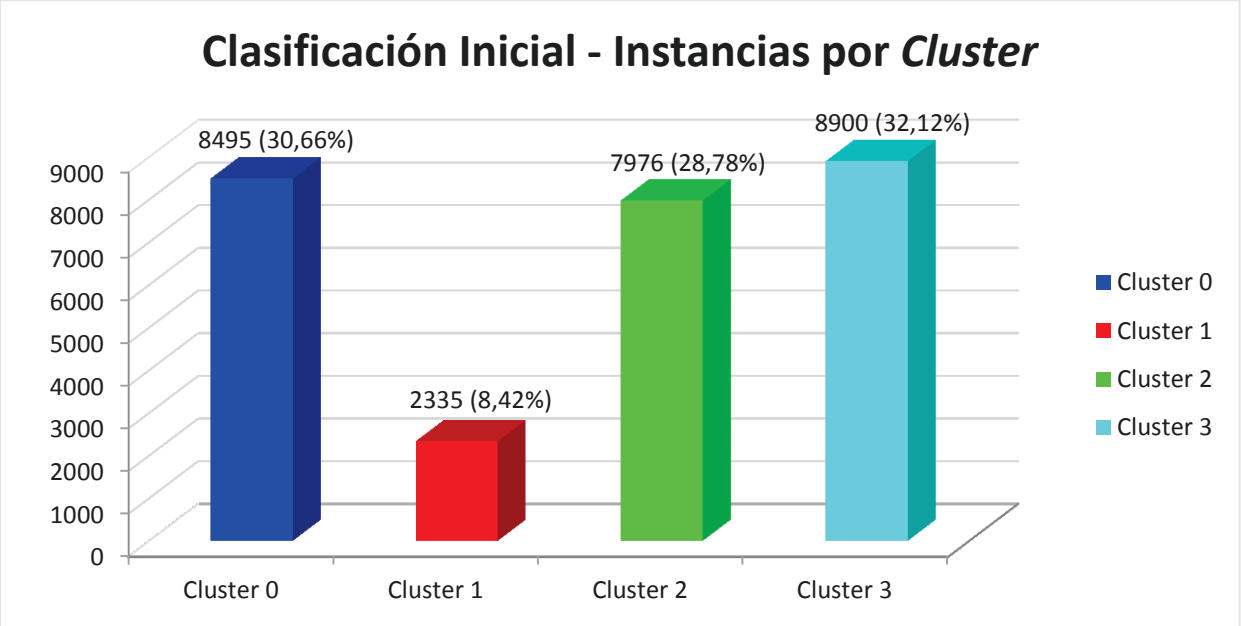
Ilustración 17: Guía de referencia para grupos de piezas

Atendiendo a este criterio, por ejemplo, el atributo SUST-44101 corresponde a la sustitución de neumáticos, debido a que está compuesto de SUST (sustitución), 44 (grupo ruedas) y 10(familia neumático). El 1 final indica que se trata de los neumáticos del lado izquierdo.

- Atributo de clase: Por último, en el último atributo de la tupla se almacena el clúster al que pertenece cada instancia. Este atributo actúa a modo de "salida deseada" como referencia en caso de realizar experimentación con algoritmos de aprendizaje supervisado. Como se aclara en el siguiente apartado, el fichero original ha sido procesado previamente para obtener este atributo.

3.5.Clasificación original

Sobre el fichero original de datos se sabe de antemano que se procesó previamente a este proyecto realizando una clasificación mediante el algoritmo de EM (Expectation Maximization), teniendo en cuenta únicamente 3 atributos: Tot_mo, Tot_pint y Tot_piez. Esta clasificación resultó en un agrupamiento de las instancias en 4 clústeres diferentes, distribuidos de la siguiente manera (Gráfica 8):



Gráfica 8: Distribución por clúster del fichero original

Esta clasificación original servirá de referencia para iniciar el proceso analítico estableciendo ciertos parámetros de base:

- Número de clústeres o categorías.
- Atributos que proporcionan la mayor ganancia de información.
- Nuevos subconjuntos de datos basados en cada clúster sobre los que aplicar *data mining.*

El número de clústeres obtenido en esta clasificación se mantendrá a lo largo de la experimentación con otros algoritmos de agrupamiento. También se considerarán 4 las categorías de severidad sobre las que se trabajará en el análisis detallado.

3.6.Preprocesado de datos

3.6.1. Normalización

Aquellos atributos que sean numéricos en el fichero deberían, por conveniencia, ser normalizados. Normalizar consiste en escalar el rango total de un atributo en el fichero (acotado por su valor máximo y su valor mínimo) al equivalente entre 0 y 1, conservando los valores de cada una de las instancias la proporcionalidad dentro del rango [0,1]. Aplicando la función de normalización siguiente se consigue el nuevo valor transformado.

$$X_{nuevo} = \frac{X_{antiguo} - X_{minimo}}{X_{maximo} - X_{minimo}}$$

La etapa de normalización es crucial, dado que pueden existir datos que se encuentren en diferentes unidades y/o escalas. Por ejemplo, a la hora de utilizar métodos de minería de datos que trabajen con la Distancia Euclídea (como el *clusterer* K-medias), es vital que los datos se encuentren en la misma escala para que la comparaciones entre ellos tengan una ponderación correcta.

Existen desventajas a la hora de aplicar esta técnica, como es el caso de que se presenten valores atípicos muy alejados de la media de los datos. Este hecho puede hacer que los verdaderos "datos útiles" se encuentren en un intervalo muy pequeño debido al escalado, pero en el conjunto de datos utilizado para este problema este fenómeno no es acusado.

Aplicar la normalización no siempre es algo necesario o deseable. En circunstancias en las que todas las variables evaluadas compartan las mismas unidades de medida, no será necesario realizar la normalización de las instancias. En grupos de atributos con las mismas unidades, normalizar los datos puede cambiar determinadas medidas, como los máximos y los promedios. A lo largo de la experimentación se indicará expresamente si los datos han sido normalizados.

WEKA ofrece un filtro que automatiza el proceso de normalización por atributo.

3.6.2. Aleatorización

También es considerado conveniente presentar las instancias de los datos de una manera aleatoria a los algoritmos de *data mining* para evitar sesgos en el aprendizaje. El conjunto de datos utilizado para el problema aparece ordenado por instancias ya que presenta el atributo denominado Instance_number que las ordena numéricamente de 0 a 27705.

WEKA ofrece un filtro que aleatoriza las instancias del fichero de datos mediante una semilla introducida por parámetros. Durante la experimentación se han utilizado diferentes semillas de manera arbitraria.

3.6.3. Eliminación de atributos superfluos

Dependiendo del experimento que se realiza en cada momento será conveniente ignorar unos atributos y mantener otros, por lo que en las secciones donde se desglosa la experimentación se han indicado las variables implicadas en cada prueba.

Sin embargo, si se puede adelantar que los atributos iniciales Instance_number, Secuencia e Historia se han obviado en todas las experimentaciones, dado que no aportan ninguna información de valor a la toma de decisiones.

La variable Tot_gen tampoco aporta información relevante en la toma de decisiones por considerarse redundante a los atributos de costes desglosados. Este atributo es la suma de los otros costes presentes en el fichero. Se mantendrá, sin embargo, como variable a observar, dado que gráficamente si puede revelar ciertos sesgos interesantes para el estudio.

4. Resultados y evaluación

4.1.Pruebas iniciales

Sobre el conjunto de datos procesado original interesa saber lo máximo posible antes de comenzar con la experimentación. Conviene realizar un análisis que revele más información acerca de cómo se ha obtenido el atributo Cluster, que sirve como "salida deseada" de cada una de las instancias del fichero original. Éste agrupamiento previo influye en gran medida en el desarrollo de la experimentación.

4.1.1. Prueba con algoritmo EM

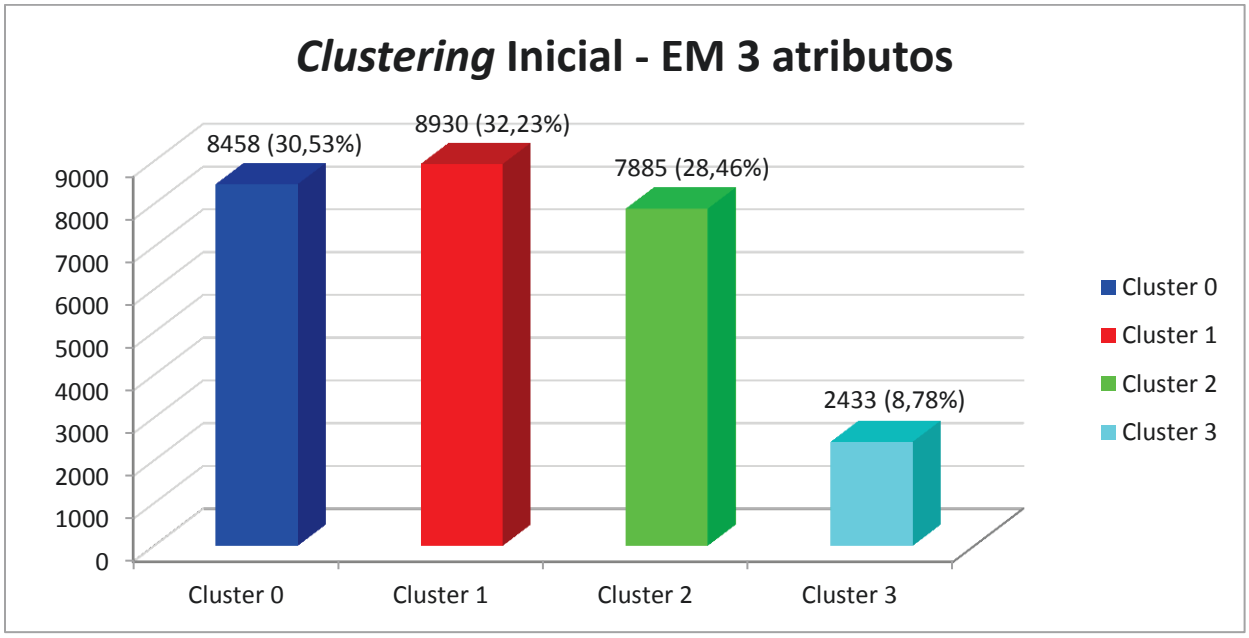
Se realiza como primera prueba un experimento con el algoritmo EM (definido en la sección 2.4.2), el mismo que fue utilizado para obtener el agrupamiento previo al proyecto, con el fin de emular el procesamiento original y extraer información al respecto. Se trata de un método de aprendizaje no supervisado, con lo que no existe entrenamiento.

Como se ha señalado anteriormente, el agrupamiento previo se realizó atendiendo sólo a 3 atributos, ignorando todos los demás:

- Tot_mo: Coste total de mano de obra por instancia.
- Tot_pint: Coste total de pintura por instancia.
- Tot_piez: Coste total de piezas sustituidas por instancia.

Debido a que todas las variables representan la misma medida (euros), para esta prueba no se ha normalizado el fichero. Sí se ha sometido a un filtro que aleatoriza las instancias.

Una vez realizado el experimento, los resultados son los siguientes (Gráfica 9):



Gráfica 9: Pruebas Iniciales - Distribución por clúster de EM

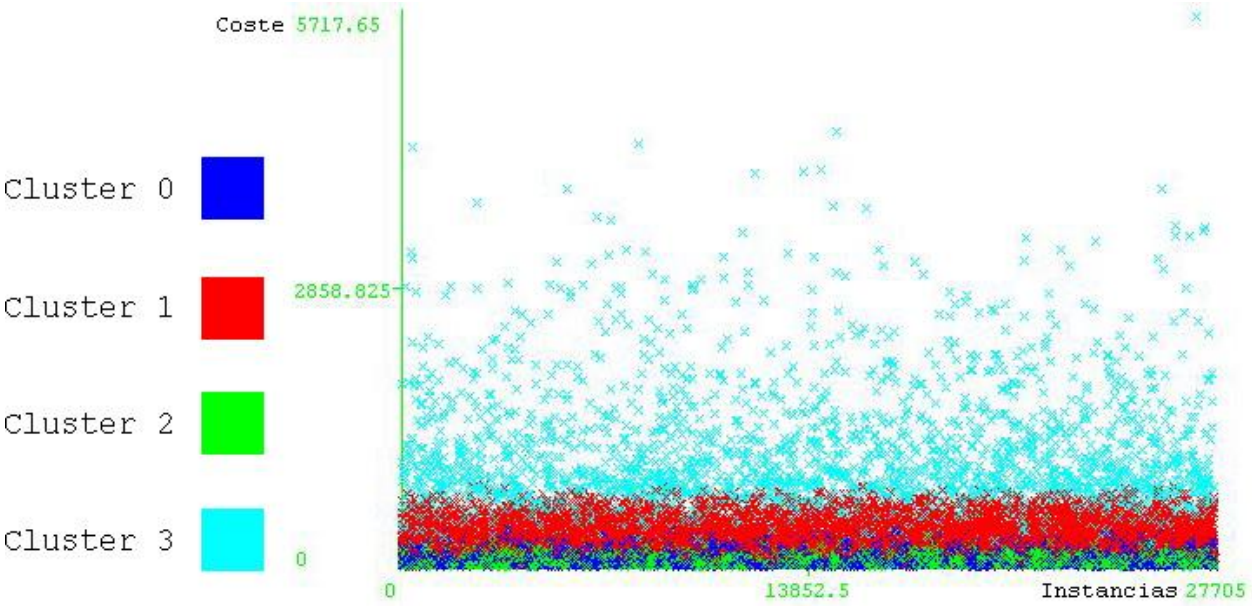
Los datos de la gráfica se asemejan sobremanera a los datos representados en la Gráfica 8, y revelan una disposición prácticamente idéntica a la que se obtuvo en el análisis previo al proyecto, como era de suponer. Puede apreciarse un intercambio en los porcentajes entre los clúster 1 y 3 debido a que el algoritmo ha nombrado los grupos de una manera diferente. La posible causa es la selección de instancias en distinto orden (han sido aleatorizadas), y por ello han aparecido los grupos en orden diferente.

```
Scheme:       weka.clusterers.EM -I 100 -N -1 -X 10 -max -1 -ll-cv 1.0E-6 -ll-iter
1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 7
Instances:    27706
Attributes:   3
              Tot_mo
              Tot_pint
              Tot_piez

EM
==
Number of clusters selected by cross validation: 4

              Cluster
Attribute            0         1         2         3
                 (0.3)     (0.32)    (0.29)    (0.09)
=========================================================
Tot_mo
  mean          94.1732  291.3244   71.9701 1021.7006
  std. dev.     63.3769  146.3705   37.6256  641.8641

Tot_pint
  mean         193.0904  460.8249  105.9655  622.5607
  std. dev.     87.9042  296.0611   92.3034  373.5388

Tot_piez
  mean           4.9288  383.1609  170.9071 2628.2235
  std. dev.      5.5708  380.8954   99.3883 2736.0727

=== Model and evaluation on training set ===
Clustered Instances

0       8458 ( 31%)
1       8930 ( 32%)
2       7885 ( 28%)
3       2433 (  9%)
```

Tabla 3: Pruebas Iniciales - Salida de EM

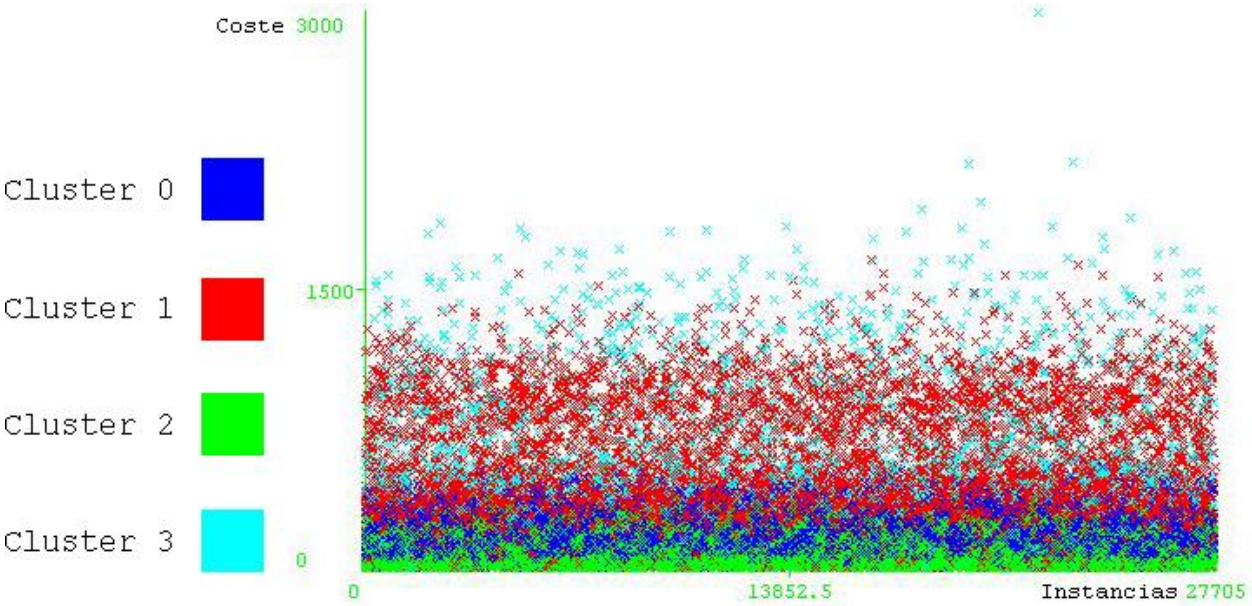
Según la salida de WEKA (Tabla 3), se puede observar que la organización por clústeres se basa en una división del rango de los atributos por tramos, siendo los costes más bajos pertenecientes a un clúster¸ y los sucesivamente más altos a otros. Desgraciadamente, no es posible a simple vista apreciar qué atributos afectan más o menos a la decisión de pertenecer a determinado clúster, o dicho de otra manera, que atributo aporta más ganancia de información.

En las siguientes gráficas se reflejan las asignaciones de cada instancia por clúster, con la intención de poder encontrar algún tipo de tendencia o patrón en la distribución de los datos.



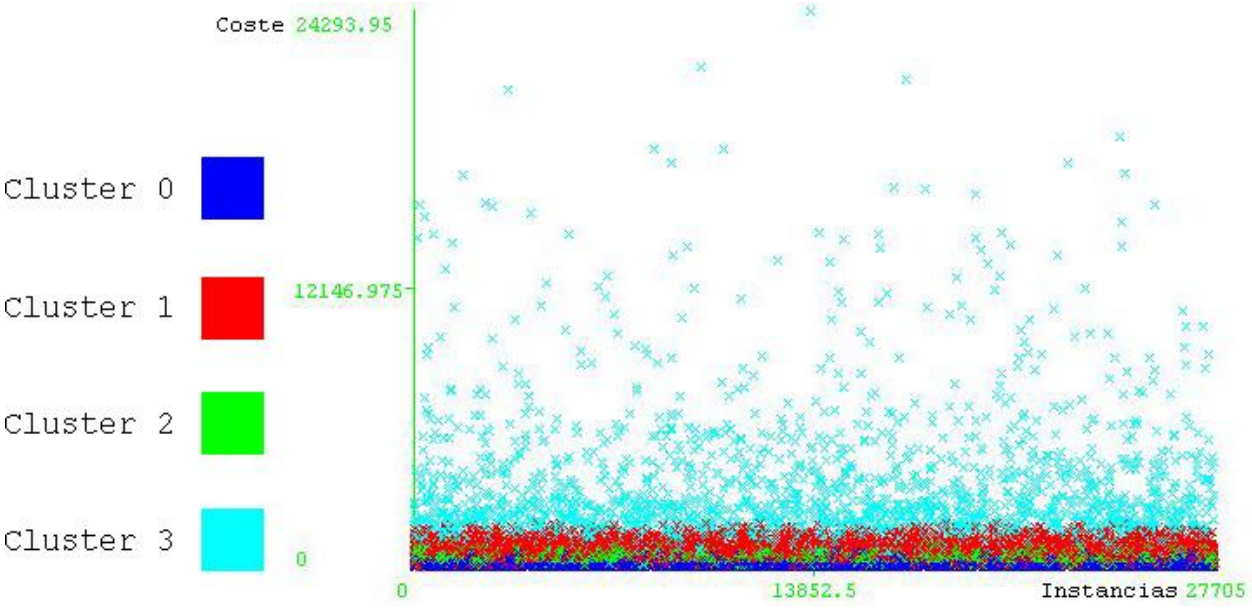
Gráfica 10: Pruebas iniciales – Distribución del atributo Tot_mo. EM

En la Gráfica 10 se puede apreciar como la distribución por clúster respecto al atributo Tot_mo establece una estratificación de las instancias. Pese a que se observan ejemplos pertenecientes a diferentes clústeres entremezclados (clústeres 0 y 2), la tónica es que los clústeres con la media de coste más baja se organicen en la parte baja de la gráfica, mientras que los que acumulan un coste más alto se sitúan en la parte superior.



Gráfica 11: Pruebas iniciales – Distribución del atributo Tot_pint. EM

La distribución del atributo Tot_pint (Gráfica 11) se encuentra menos diferenciada que en el caso anterior, esto quizá sea un indicativo de que este atributo Tot_pint presenta una peor ganancia de información que el atributo Tot_mo anterior.



Gráfica 12: Pruebas iniciales – Distribución del atributo Tot_piez. EM

En esta última Gráfica 12 se aprecia también estratificación. Pese a estar "comprimida" debido a la existencia de datos atípicos en la parte superior de la gráfica, la parte inferior presenta una estructura organizada, con cada clúster formando una capa. Coincide además que los clústeres se encuentran ordenados de menor a mayor media de coste en la gráfica.

Quizá éste último atributo sea el que mayor influye en la decisión de agrupamiento, aunque éstas pruebas no son concluyentes. En subsiguientes pruebas se va a intentar contrastar este hecho.

La asignación que ha realizado el EM de cada instancia a cada clúster se ha salvado en un fichero de tipo ARFF para posterior experimentación.

4.1.2. Prueba con algoritmo K-medias

Es interesante comprobar si otros algoritmos de *clustering* también tienden a realizar esta clasificación sobre el conjunto de datos. Por ello, se ha realizado una prueba con otro algoritmo de agrupamiento diferente, K-medias (sección 2.4.2). De nuevo, se trata de un método no supervisado, por lo que se realizará la prueba y se evaluará de manera gráfica. WEKA implementa el algoritmo mediante el *clusterer* SimpleKMeans.

En este caso, también se ha optado por no normalizar las instancias y utilizar los 3 atributos sobre los que se basada la agrupación original. Se ha establecido el número de clústeres a encontrar por el algoritmo en 4 y se han dejado el resto de parámetros por defecto.

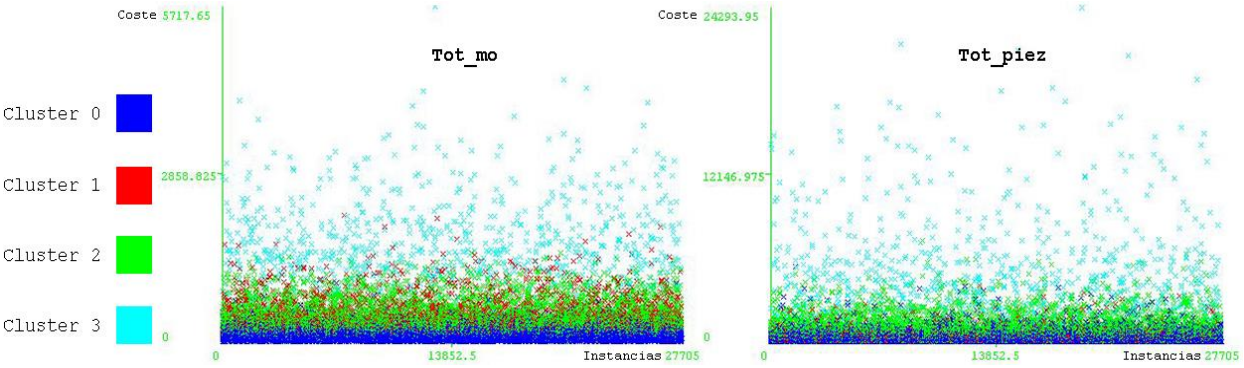
Los resultados arrojados por la prueba con K-medias son los siguientes (Tabla 4: Pruebas iniciales – Salida de K-medias):

```
Scheme:          weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-
pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 4 -A "weka.core.EuclideanDistance
-R first-last" -I 100 -num-slots 1 -S 12
Instances:    27706
Attributes:   3
              Tot_mo
              Tot_pint
              Tot_piez

kMeans
======
Number of iterations: 26
Within cluster sum of squared errors: 104.76169281423357

Initial starting points (random):

Cluster 0: 112,204,269.85
Cluster 1: 846,894.77,1495.47
Cluster 2: 586.33,455.55,2195.27
Cluster 3: 1065.94,1208.96,1051.38

Final cluster centroids:
                        Cluster#
Attribute   Full Data          0          1          2          3
            (27706.0) (17349.0)   (2521.0)   (7052.0)    (784.0)
============================================================
Tot_mo       237.2872   93.7908   443.176  356.0192 1682.6651
Tot_pint     294.3088  138.5351  968.3477  391.1563  702.8531
Tot_piez     417.4981  155.9324  239.3102  599.9726 5137.2726

=== Model and evaluation on training set ===
Clustered Instances

0      17349 ( 63%)
1       2521 (  9%)
2       7052 ( 25%)
3        784 (  3%)
```

Tabla 4: Pruebas iniciales – Salida de K-medias

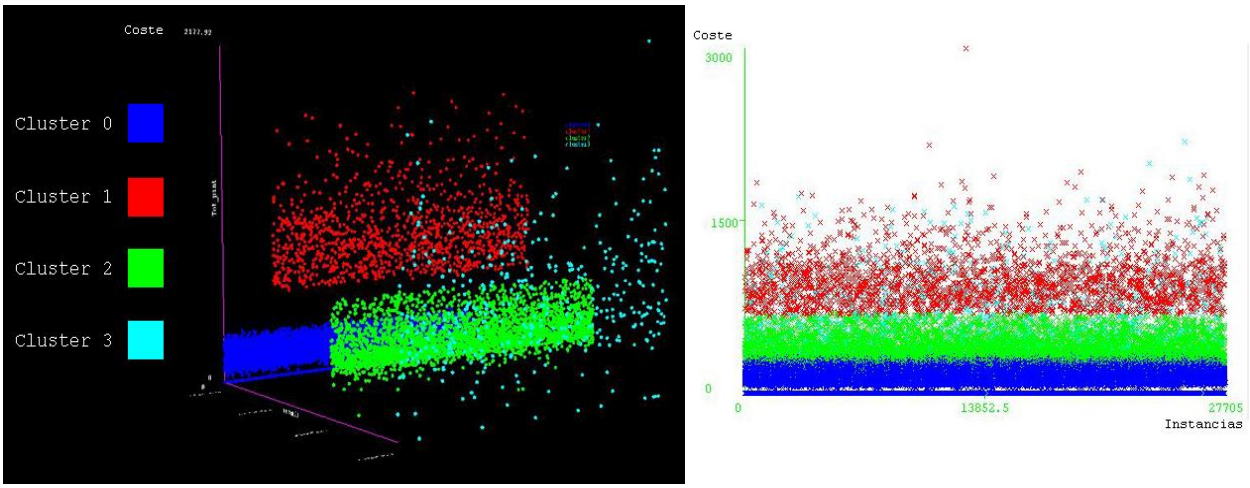
A primera vista, los datos revelan una distribución absolutamente diferente a la obtenida a través de EM, pero quizá investigando sobre las distribuciones de cada atributo puedan descubrirse similitudes entre ambos agrupamientos.

Las gráficas de distribución de los atributos Tot_mo o Tot_piez (Gráfica 13) presentan una tendencia hacia la estratificación pero no llegan a delimitar claramente los clústeres, debido a que demasiadas instancias aparecen entremezcladas y no existe una delimitación clara entre grupo y grupo. Estos atributos no aportan demasiada información al modelo.



Gráfica 13: Pruebas iniciales – Distribución de los atributos Tot_mo y Tot_piez. K-medias

Sin embargo, la gráfica con el atributo Tot_pint (Gráfica 14) revela una estratificación prácticamente prefecta, con una separación entre los clústeres 0, 2 y 1 muy marcada. El clúster número 3 casi podría considerarse residual, dado que aglutina muy pocas instancias (apenas el 3%), es probable que en otras ejecuciones con distinta semilla aleatoria pueda incluso llegar a desaparecer. Al forzar la ejecución de K-medias con 3 clústeres únicamente, se constata que desaparece ese último grupo, quedando los porcentajes de los otros apenas alterados.



Gráfica 14: Pruebas iniciales – Distribución del atributo Tot_pint. K-medias

Si se compara con detenimiento esta distribución de grupos con la generada por el algoritmo EM, puede apreciarse una tendencia del algoritmo K-medias a aglutinar en un único clúster las instancias que en EM estarían aglutinadas en los clústeres 0 y 2. Por tanto, si desechamos el clúster del 3%, 63% sería más o menos la suma de los 31% y 28% correspondientes a los clústeres 0 y 2. Los otros porcentajes de 25% y 9% restantes corresponden a los clústeres 1 y 3 del EM.

Esta clasificación, sea como fuere, se considera como alternativa a la anteriormente establecida con EM, en el apartado 4.1.4 se explicará con más detalle esta propuesta. Este análisis aún no se considera concluyente, por lo que se seguirán haciendo otras pruebas que proporcionen más información.

4.1.3. Clasificación con OneR

Para poder corroborar el estudio de la ganancia de información de cada atributo, se someterá al conjunto de datos a un experimento con el algoritmo OneR (expuesto en la sección 2.4.1). Se utilizará la asignación de clústeres derivada de la prueba con EM.

El algoritmo OneR se considera el algoritmo supervisado de clasificación más sencillo, y crea una regla de decisión con cada variable de entrada del conjunto. De todas éstas, elegirá aquella que devuelva el error más pequeño de clasificación. Este algoritmo es útil para determinar qué atributos influyen más en la decisión, es decir, que atributos tienen una mejor ganancia de información.

La prueba consistirá en ejecutar el algoritmo OneR tantas veces como atributos haya. Dado que únicamente se están utilizando 3 atributos para el agrupamiento (la salida deseada, Cluster, no cuenta), el algoritmo se ejecutará 3 veces. Sucesivamente, el algoritmo irá devolviendo un conjunto de reglas basadas en un único atributo, que será el que aporte mejor ganancia de información en ese momento. Se procederá a ignorar ese atributo y analizar de nuevo el conjunto de datos, hasta que ya no queden más atributos por descartar.

Los resultados de la prueba se pueden apreciar en las siguientes tablas (Tabla 5, Tabla 6 y Tabla 77):

```
Scheme:       weka.classifiers.rules.OneR -B 10
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster

=== Classifier model (full training set) ===
Tot_piez:
       < 14.22      -> cluster0
       < 14.434999999999999      -> cluster1
       < 16.29      -> cluster0
        . . .

=== Summary ===
Correctly Classified Instances       21445               77.402  %
Incorrectly Classified Instances      6261               22.598  %
Kappa statistic                          0.6815
Mean absolute error                      0.113
Root mean squared error                  0.3361
Relative absolute error                 31.6401 %
Root relative squared error             79.5492 %
Coverage of cases (0.95 level)          77.402  %
Mean rel. region size (0.95 level)      25      %
Total Number of Instances            27706

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8312  103   43    0 |    a = cluster0
  834 5478 2540   78 |    b = cluster1
   52 1774 6059    0 |    c = cluster2
   12  657  168 1596 |    d = cluster3
```

Tabla 5: Pruebas iniciales - Primera prueba OneR

En la primera pasada se observa que el algoritmo elige el atributo Tot_piez para el conjunto *one rule*, es decir, que ese atributo es el que mejor resultado de clasificación obtiene. Se puede apreciar además, que el resultado no es del todo malo (77%) para tratarse de una única regla de decisión. Cabe destacar que el atributo Cluster se mantiene durante todas las pasadas como referencia para la supervisión. No se han plasmado tampoco todas las reglas generadas por el algoritmo (se indica con unos puntos suspensivos) porque en esta prueba interesa más el atributo escogido que el modelo creado.

```
Scheme:       weka.classifiers.rules.OneR -B 10
Instances:    27706
Attributes:   3
              Tot_mo
              Tot_pint
              Cluster

=== Classifier model (full training set) ===
Tot_mo:
       < 1.33        -> cluster0
       < 5.465       -> cluster2
       < 6.02        -> cluster0
       . . .

=== Summary ===
Correctly Classified Instances       17129               61.8242 %
Incorrectly Classified Instances     10577               38.1758 %
Kappa statistic                          0.4621
Mean absolute error                      0.1909
Root mean squared error                  0.4369
Relative absolute error                 53.4511 %
Root relative squared error            103.3939 %
Coverage of cases (0.95 level)          61.8242 %
Mean rel. region size (0.95 level)      25      %
Total Number of Instances            27706

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 3510 1450 3498    0 |    a = cluster0
  924 7458  417  131 |    b = cluster1
 3237  250 4398    0 |    c = cluster2
   19  646    5 1763 |    d = cluster3
```

Tabla 6: Pruebas iniciales- Segunda prueba OneR

En la segunda pasada, ignorando el atributo Tot_piez, se observa que el algoritmo escoge en este caso Tot_mo para la clasificación. El resultado de clasificación es sensiblemente peor (61%), es decir, este atributo ha influido menos que el anterior en la decisión de agrupamiento.

```
Scheme:       weka.classifiers.rules.OneR -B 10
Instances:    27706
Attributes:   2
              Tot_pint
              Cluster

=== Classifier model (full training set) ===
Tot_pint:
       < 50.275      -> cluster2
       < 54.96       -> cluster0
       < 61.855000000000004       -> cluster2
```

```
       . . .

=== Summary ===
Correctly Classified Instances       15598               56.2983 %
Incorrectly Classified Instances     12108               43.7017 %
Kappa statistic                          0.3745
Mean absolute error                      0.2185
Root mean squared error                  0.4674
Relative absolute error                 61.188  %
Root relative squared error            110.6242 %
Coverage of cases (0.95 level)          56.2983 %
Mean rel. region size (0.95 level)      25      %
Total Number of Instances            27706

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 5089 1587 1770   12 |    a = cluster0
 1884 6089  635  322 |    b = cluster1
 3347  442 4096    0 |    c = cluster2
  180 1835   94  324 |    d = cluster3
```

Tabla 7: Pruebas iniciales - Tercera prueba OneR

Por último, solo queda Tot_pint, el atributo con peor ganancia de información. De los 3 atributos utilizados para el agrupamiento, es el que menos aporta a la decisión.

Contrastando éstos resultados con los de las anteriores gráficas de distribución (Gráfica 10, Gráfica 11 y Gráfica 12), se puede corroborar lo siguiente:

Los atributos utilizados para la clasificación, ordenados de mejor a peor ganancia de información son:

$$\boldsymbol{Tot\_piez > Tot\_mo > Tot\_pint}$$

Por tanto, el factor más determinante para la decisión de agrupación es el coste total de piezas, seguido del coste de mano de obra y el coste de operaciones de pintura.

4.1.4. Definición de categorías:

Antes de realizar más experimentos, también es interesante analizar la variable Tot_gen, no incluida en el algoritmo de clasificación pero un reflejo de la tendencia de agrupación que ha tomado el algoritmo.

Tot_gen es el coste total de una reparación, incluyendo el coste de pintura, mano de obra y piezas, por tanto, es un buen parámetro sobre el que ordenar o agrupar las instancias. La severidad de un siniestro se puede medir mediante el coste total de las reparaciones que deben realizarse sobre el vehículo.

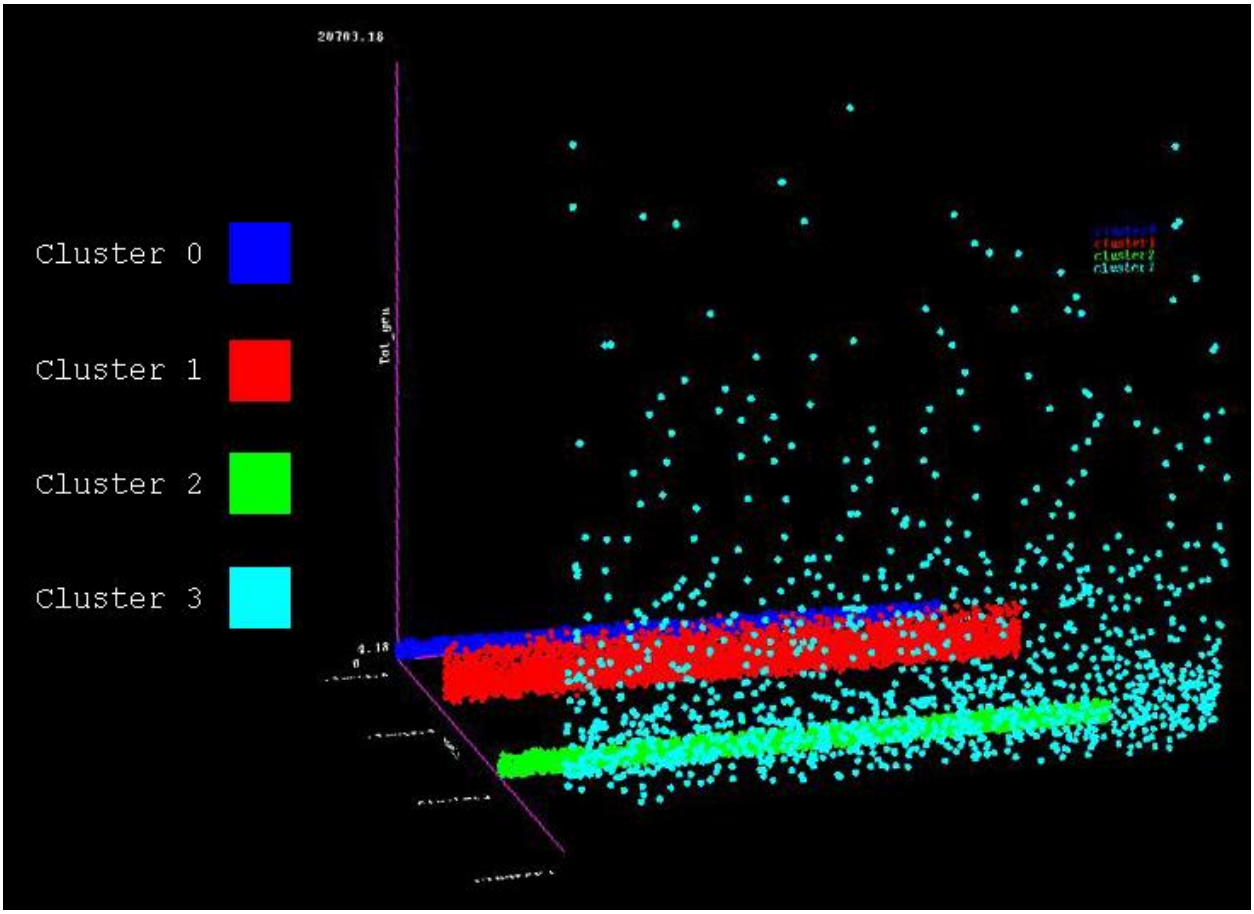
No se incluye sin embargo en los algoritmos porque es simplemente la suma de los otros costes, se considera redundante y no aporta nueva información que no hayan aportado ya las otras variables de coste.

En la siguiente tabla se recogen diferentes características del atributo en cada uno de los diferentes clústeres (Tabla 8):

	Tot_gen			
	Cluster 0	**Cluster 1**	**Cluster 2**	**Cluster 3**
Media	291,35	1154,14	351,95	4425,14
Desviación típica	134,44	436,66	155,55	2999,52
Valor máximo	731,42	2708,64	780,90	25525,60
Valor mínimo	4,18	279,93	19,20	1097,60

Tabla 8: Pruebas iniciales - Características de Tot_gen

Los valores estadísticos presentados en la tabla establecen una clara delimitación entre los diferentes clústeres. Se puede apreciar que atendiendo a las medias, los grupos se distribuyen, ordenados de menor a mayor media, como Cluster 0, Cluster 2, Cluster 1 y Cluster 3. Todas las otras medidas estadísticas también cumplen esta regla, el menor mínimo corresponde también al clúster con menor máximo, menor media y menor desviación típica, y así sucesivamente. Si observamos además la proyección tridimensional (Gráfica 15), también se constata la estratificación anteriormente mencionada.



Gráfica 15: Pruebas iniciales - Proyección 3D de la distribución de Tot_gen

La distribución es bastante parecida a la del atributo Tot_piez (Gráfica 12), con los clústeres 0 y 2 bastante solapados, y por encima el 1 y 3, respectivamente.

Se puede por tanto concluir que la agrupación original atiende al coste del siniestro, o lo que es lo mismo, a su severidad. Cada clúster representa un grado de severidad diferente. Estos grados o categorías podrían denominarse de la siguiente manera:

- Grado 1 (clúster 0): **Severidad leve**.
- Grado 2 (clúster 2): **Severidad moderada**.
- Grado 3 (clúster 1): **Severidad alta.**
- Grado 4 (clúster 3): **Severidad muy alta.**

La categoría más habitual en porcentaje (32%) es el grado 3 (severidad alta), seguido muy de cerca por los grados 1 (31%) y 2 (28%), y por último el menos habitual grado 4, que sólo presenta un 9%.

Como se ha comentado en el análisis mediante el algoritmo K-medias, en ocasiones la diferencia entre grados 1 y 2 para una determinada instancia no está claramente delimitada, por lo que, complementando esta primera clasificación, podría sugerirse una segunda más simplificada que combinara los grados 1 y 2 en un único grupo:

- Grado 1 (clústeres 0 y 2): **Severidad moderada.**
- Grado 2 (clúster 1): **Severidad moderada.**
- Grado 3 (clúster 3): **Severidad muy alta.**

En principio, se utilizará la primera clasificación a lo largo de toda la experimentación.

4.2.Análisis de severidad:

Como se menciona en la sección 2.2.2, uno de los posibles riesgos de fraude radica en la propia tasación de los daños realizada por los proveedores para la compañía aseguradora. Un siniestro puede ser valorado de manera errónea, suponiendo un desajuste (tanto a la baja como al alza) en el presupuesto que se debe realizar para la reparación. De la misma manera, el presupuesto que presenta el taller para la reparación puede estar hinchado para obtener mayor beneficio de la operación. En cualquiera de los casos es deseable contar con mecanismos que permitan contrastar el informe de valoración y detectar anomalías en caso de que existan.

Una de las aproximaciones que propone este proyecto es la de diseñar un modelo de conocimiento que, atendiendo a los grados de severidad establecidos en la sección anterior, permita clasificar nuevos siniestros mediante un conjunto de reglas de decisión de manera automática.

De esta manera, con este sistema puede catalogarse un siniestro dentro de un grado de severidad (representado por un clúster) de forma automática. Se puede contrastar un presupuesto o tasación aplicándole las reglas de decisión obtenidas y verificando si se cataloga en el grado de severidad correcto (Ilustración 18).

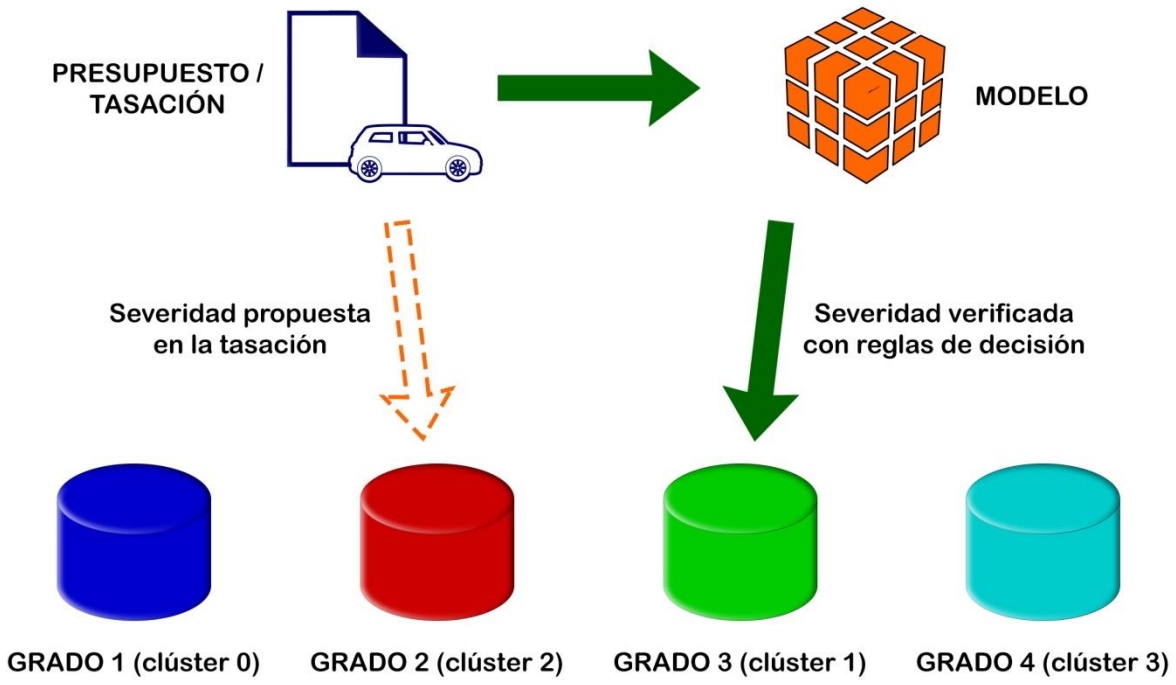


Ilustración 18: Análisis de tasación

4.2.1. Primer modelo con árboles de decisión

Como primera opción para la obtención de un conjunto de reglas de decisión se ha optado por someter al conjunto de datos a una prueba con el clasificador C4.5, el cual genera un árbol de decisión y un conjunto de reglas asociadas.

Pese a que el conjunto de datos inicial ya contaba con un atributo de clase Cluster, en los siguientes experimentos se ha optado por utilizar el atributo de clase obtenido de la primera prueba con el *clusterer* EM, dado que difiere ligeramente en algunas instancias con respecto al primero y el etiquetado de clústeres tiene un orden distinto. El cambio no es sustancial, pero se considera más coherente utilizar en todo momento el atributo de clase obtenido en los experimentos. Éste segundo atributo de clase también se llama Cluster debido a que el algoritmo EM lo nombra de esta manera automáticamente.

La primera prueba consiste en ejecutar el clasificador J48 (C4.5) sobre el conjunto de datos. El árbol C4.5 se basa en el ratio de ganancia de información para designar las bifurcaciones, por lo tanto los atributos con mayor ganancia se posicionarán en los nodos más altos o raíz, mientras que aquellos que aportan menos a la decisión se situarán cercanos a las hojas.

Siguiendo la tónica del experimento con EM, sólo se tendrán en cuenta los 3 atributos que reflejan los costes, que son los siguientes:

- Tot_mo: Coste total de mano de obra por instancia.
- Tot_pint: Coste total de pintura por instancia.
- Tot_piez: Coste total de piezas sustituidas por instancia.

Para esta prueba no se ha considerado necesario normalizar los datos, aunque sí que han sido sometidos a un filtro aleatorizador. Se aplica validación cruzada k=10 en todos los experimentos.

El clasificador J48 permite dos tipos de poda del árbol. La primera se basa en el factor de confianza (por defecto), y la segunda consiste en aplicar poda de error reducido (r*educed error pruning*, REP). La poda de error reducido funciona de la siguiente manera: empezando desde las hojas, cada nodo es sustituido por su clase más común. Si el error de predicción no se ve afectado, se mantiene dicho cambio [43]. De esta manera se reduce la dimensión del árbol de manera más rápida que con el factor de confianza.

Para la primera pasada se han utilizado las configuraciones por defecto en WEKA para ambos tipos de ejecución con poda, con un factor de confianza de 0.25 y número mínimo de instancias por hoja (minNumObj) de 2, y en el caso del *reduced error pruning*, un minNumObj también de 2 y número de *folds* para la poda igual a 3.

Los resultados de este experimento se reflejan en la Tabla 9. En todas las pruebas que se realicen con el clasificador J48 se reflejarán los resultados con los dos métodos de poda mencionados. Al final de la sección se muestra una comparativa (Tabla 13) de todas las pruebas realizadas.

```
Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
(...)

Number of Leaves  :         157
Size of the tree :          313

=== Summary ===
Correctly Classified Instances       27394               98.8739 %
Incorrectly Classified Instances       312                1.1261 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8395   36   27    0 |    a = cluster0
   34 8801   63   32 |    b = cluster1
   21   67 7797    0 |    c = cluster2
    0   32    0 2401 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 7 -M 2
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
(...)

Number of Leaves  :         95
Size of the tree :          189

=== Summary ===
Correctly Classified Instances       27320               98.6068 %
Incorrectly Classified Instances       386                1.3932 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8385   45   28    0 |    a = cluster0
   49 8764   83   34 |    b = cluster1
   26   80 7779    0 |    c = cluster2
    0   41    0 2392 |    d = cluster3
```

Tabla 9: Análisis de tasaciones – J48 primer modelo. Prueba 1. CF y REP

Esta primera pasada revela unos resultados muy buenos en cuanto al porcentaje de clasificación, con casi un 99%, pero plantea serios problemas debido al desmesurado tamaño del árbol en ambas alternativas, lo que hace suponer que con casi toda seguridad que el modelo presente *overfitting* a los datos de entrenamiento. Se ha optado por no incluir el árbol o las reglas asociadas debido a que ocupan demasiado espacio.

Para paliar estas desventajas, se cambiarán los parámetros para forzar la post-poda de una manera mucho más acentuada y sin apenas penalización. Un árbol más sencillo generalizará mejor y será más manejable. Reduciendo el factor de confianza y aumentando el número mínimo de instancias por hoja se pueden conseguir mejores resultados en la poda con CF, mientras que en la poda con REP, basta con ampliar el número mínimo de instancias por hoja.

En una segunda pasada, reduciendo el factor de confianza (CF) a 0.05 los resultados son mucho mejores, como refleja la Tabla 10. En ambas alternativas de poda se ha aumentado el minNumObj a 50.

```
Scheme:       weka.classifiers.trees.J48 -C 0.05 -M 50
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
(...)

Number of Leaves  :         33
Size of the tree :          65

=== Summary ===
Correctly Classified Instances       27059               97.6648 %
Incorrectly Classified Instances       647                2.3352 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8290   90   78    0 |    a = cluster0
   73 8642  117   98 |    b = cluster1
   19   85 7781    0 |    c = cluster2
    0   87    0 2346 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 34 -M 50
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
(...)

Number of Leaves  :         18
Size of the tree :          35

=== Summary ===
Correctly Classified Instances       26979               97.376  %
Incorrectly Classified Instances       727                2.624  %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8310   78   70    0 |    a = cluster0
  103 8599  133   95 |    b = cluster1
   47   86 7752    0 |    c = cluster2
    0  115    0 2318 |    d = cluster3
```

Tabla 10: Análisis de tasaciones – J48 primer modelo. Prueba 2. CF y REP

Como se puede comprobar, la penalización en el porcentaje de clasificación es apenas inexistente, y el tamaño del árbol es muchísimo más pequeño, lo que permite una mejor generalización. Es destacable la mayor capacidad de poda que presenta el REP, con una precisión prácticamente idéntica pero un árbol sensiblemente más pequeño. Pese a todo, aún es posible tener un mejor resultado si se siguen afinando los parámetros del algoritmo antes mencionados. Simplemente aumentando el "minNumObj" de 50 a 100 en ambas alternativas, el árbol todavía se puede reducir más, manteniendo un porcentaje de clasificación todavía muy alto (Tabla 11).

Para la alternativa con poda CF, se ha reducido el factor de confianza a 0.01.

```
Scheme:       weka.classifiers.trees.J48 -C 0.01 -M 100
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
(...)

Number of Leaves  :         24
Size of the tree :          47

=== Summary ===
Correctly Classified Instances       26930               97.1992 %
Incorrectly Classified Instances       776                2.8008 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8301   96   61    0 |    a = cluster0
  104 8571  148  107 |    b = cluster1
   50   84 7751    0 |    c = cluster2
    0  126    0 2307 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 34 -M 100
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
(...)

Number of Leaves  :         14
Size of the tree :          27

=== Summary ===
Correctly Classified Instances       26834               96.8527 %
Incorrectly Classified Instances       872                3.1473 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8257  144   57    0 |    a = cluster0
   90 8655  105   80 |    b = cluster1
   53  187 7645    0 |    c = cluster2
    0  156    0 2277 |    d = cluster3
```

Tabla 11: Análisis de tasaciones – J48 primer modelo. Prueba 3. CF y REP

Se puede continuar simplificando el modelo, mientras no pierda precisión, hasta llegar a un punto en el cual el tamaño del mismo, el número de reglas y su porcentaje de clasificación se consideren razonables. En la siguiente prueba se ha conseguido descender el tamaño del árbol a 25 y 23 nodos para CF y REP, respectivamente. Habiendo alcanzado un número mínimo de instancias por hoja de 150, y en la primera poda, reduciendo el CF hasta 0.005. La capacidad de clasificación del algoritmo sólo se ha visto mermada en un 1% con respecto al anterior modelo probado en ambos casos.

Sería posible seguir ampliando el mínimo número de instancias por hoja, reduciendo el árbol mucho más, pero se ha considerado conveniente parar en este punto. Los árboles generados son suficientemente manejables, claros y sencillos (Tabla 12).

```
Scheme:       weka.classifiers.trees.J48 -C 0.005 -M 150
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
(...)

Number of Leaves  :         13
Size of the tree :          25

=== Summary ===
Correctly Classified Instances       26836               96.8599 %
Incorrectly Classified Instances       870                3.1401 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8269  142   47    0 |    a = cluster0
   93 8688   64   85 |    b = cluster1
   55  230 7600    0 |    c = cluster2
    0  154    0 2279 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 5 -M 150
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
(...)

Number of Leaves  :         12
Size of the tree :          23

=== Summary ===
Correctly Classified Instances       26673               96.2716 %
Incorrectly Classified Instances      1033                3.7284 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8183  232   43    0 |    a = cluster0
   64 8616  160   90 |    b = cluster1
   60  230 7595    0 |    c = cluster2
    0  154    0 2279 |    d = cluster3
```

Tabla 12: Análisis de tasaciones - J48 primer modelo. Prueba 4. CF y REP

A la hora de decantarse por uno de los dos modelos, es preferible optar por aquel generado mediante poda REP, con un árbol más reducido.

En la siguiente comparativa se puede observar la evolución de las pruebas J48 (Tabla 13Tabla 13):

Prueba	CF				REP			
	CF	minNumObj	Tamaño	Precisión	Folds	minNumObj	Tamaño	Precisión
1	0.25	2	313	98,87%	3	2	189	98,61%
2	0.05	50	65	97,66%	3	50	35	97,38%
3	0.01	100	47	97,20%	3	100	27	96,85%
4	0.005	150	25	96,86%	3	150	23	96,27%

Tabla 13: Análisis de tasaciones – J48 primer modelo. Comparativa de pruebas

El modelo elegido es el obtenido por REP en la prueba número 4, con un minNumObj de 150 y número de *folds* igual a 3.

El porcentaje de instancias bien clasificadas en el mejor modelo es muy alto, aun habiendo obtenido un árbol no muy intrincado. Este hecho se debe principalmente a la baja dimensionalidad del conjunto probado. Teniendo sólo 3 atributos, se pueden generar modelos muy precisos a la par que sencillos. Las reglas de decisión asociadas al modelo se pueden consultar en la Tabla 14. La salida completa y el árbol generado por WEKA están disponibles en el ANEXO C: Salidas de WEKA.

```
J48 pruned tree
------------------
Tot_piez <= 21.01
|   Tot_pint <= 381.34
|   |   Tot_mo <= 283.3: cluster0 (5426.0/53.0)
|   |   Tot_mo > 283.3: cluster1 (154.0/53.0)
|   Tot_pint > 381.34
|   |   Tot_pint <= 527.73
|   |   |   Tot_mo <= 237.85: cluster0 (223.0/60.0)
|   |   |   Tot_mo > 237.85: cluster1 (152.0/6.0)
|   |   Tot_pint > 527.73: cluster1 (309.0/4.0)
Tot_piez > 21.01
|   Tot_mo <= 164.25
|   |   Tot_piez <= 437.21
|   |   |   Tot_pint <= 279.88: cluster2 (5027.0/63.0)
|   |   |   Tot_pint > 279.88
|   |   |   |   Tot_pint <= 363.43: cluster2 (259.0/106.0)
|   |   |   |   Tot_pint > 363.43: cluster1 (268.0/3.0)
|   |   Tot_piez > 437.21: cluster1 (553.0/66.0)
|   Tot_mo > 164.25
|   |   Tot_mo <= 709.59
|   |   |   Tot_piez <= 1584.08: cluster1 (4530.0/131.0)
|   |   |   Tot_piez > 1584.08: cluster3 (401.0/21.0)
|   |   Tot_mo > 709.59: cluster3 (1169.0/22.0)
```

Tabla 14: Análisis de tasaciones – J48 primer modelo. Reglas de decisión asociadas al mejor modelo

El análisis del árbol obtenido además corrobora más si cabe lo descubierto en la sección 4.1, situando aquellos atributos con mayor ganancia de información más cerca de la raíz del árbol. Se observa como Tot_piez es el atributo con mayor ganancia, como se concluía de las pruebas iniciales, y que lo siguen Tot_mo y Tot_pint.

El modelo conseguido se considera excelente en porcentaje de clasificación y sencillo, sin embargo, presenta algunas debilidades: parece incompleto, dado que sólo se ha basado en 3 atributos cuando el *dataset* original cuenta con 115, y además sólo está basado en costes generales, que se ven afectados por fluctuaciones del mercado, diferencia de precios entre marcas e inflación anual. Estos puntos débiles hacen que el modelo pierda fiabilidad, robustez y polivalencia, con lo que es conveniente continuar el análisis para intentar obtener un modelo más completo.

4.2.2. Ampliación del espectro

En vista de los objetivos de este proyecto, el modelo obtenido en el apartado anterior se considera insuficiente. Además de las razones argumentadas en el apartado anterior, un modelo basado exclusivamente en los costes totales es fácil de burlar si se quiere falsear el informe de tasación, dado que estos costes no son contrastables de ninguna manera. Sin embargo, incluyendo nuevos parámetros más "tangibles", como las reparaciones, operaciones de pintura y sustituciones realizadas en el vehículo, se pueden justificar los costes mencionados y añadir robustez y fiabilidad al modelo.

Las nuevas pruebas ejecutarán de nuevo el clasificador C4.5 sobre el conjunto de datos, pero en esta ocasión se ampliará el rango de atributos incluyendo todos los atributos binarios. El objetivo de esta prueba es encontrar nuevos atributos no incluidos anteriormente que aporten ganancia de información y sean decisivos en la decisión de clasificación. En otras palabras, realizar de nuevo la prueba para encontrar si en las raíces del árbol aparecen los nuevos atributos añadidos. Se van a tener en cuenta los siguientes atributos:

- Los 3 atributos de costes: Tot_mo, Tot_pint y Tot_piez.
- Los 105 atributos binarios de operaciones de pintura, reparación y sustitución.
- Los 2 atributos de conteo de piezas reparadas y sustituidas: Pos_int y Pos_mod.

En esta ocasión, dado que los grupos de atributos manejan unidades diferentes, se ha optado por normalizar los datos además de la aplicación del habitual filtro de aleatorización. También se aplica validación cruzada k=10 y se presentan dos alternativas de poda diferenciadas, CF y REP.

Para la primera pasada se han utilizado los parámetros por defecto en WEKA para la poda con CF, con un factor de confianza de 0.25 y minNumObj de 2. Para el *reduced error pruning*, se utiliza también un minNumObj de 2 y un número de *folds* igual a 3.

```
Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Instances:    27706
Attributes:   111
(...)

Number of Leaves  :         156
Size of the tree :         311

=== Summary ===
Correctly Classified Instances       27286               98.4841 %
Incorrectly Classified Instances       420                1.5159 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8385   41   32    0 |    a = cluster0
   49 8755   85   41 |    b = cluster1
   39   79 7767    0 |    c = cluster2
    0   54    0 2379 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 12 -M 2
Instances:    27706
Attributes:   111

Number of Leaves  :         100
Size of the tree :         199
```

```
=== Summary ===
Correctly Classified Instances       27198               98.1665 %
Incorrectly Classified Instances       508                1.8335 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8369   47   42    0 |    a = cluster0
   62 8715  101   52 |    b = cluster1
   52   91 7742    0 |    c = cluster2
    0   61    0 2372 |    d = cluster3
```

Tabla 15:Análisis de tasaciones - J48 ampliado. Prueba 1. CF y REP

Los resultados de la Tabla 15Tabla 15 revelan un porcentaje de acierto excelente con ambas podas, pero se constata lo que ocurrió con el experimento anterior, y es que utilizando los parámetros por defecto de J48 para este conjunto de datos se obtiene un árbol muy grande que es poco manejable y probablemente no generalice como debiera. Se va a optar por cambiar la configuración a aquella que obtuvo los mejores modelos en la prueba anterior.

En esta ocasión directamente se utilizarán los parámetros de CF reducido a 0.005, minNumObj situado en 150 para ambas podas y número de *folds* igual a 3 en el caso del *reduced error pruning*.

```
Scheme:       weka.classifiers.trees.J48 -C 0.005 -M 150
Instances:    27706
Attributes:   111
(...)

Number of Leaves  :         15
Size of the tree :         29

=== Summary ===
Correctly Classified Instances       26643               96.1633 %
Incorrectly Classified Instances      1063                3.8367 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8270  188    0    0 |    a = cluster0
   51 8606  185   88 |    b = cluster1
  212  194 7479    0 |    c = cluster2
    0  145    0 2288 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 5 -M 150
Instances:    27706
Attributes:   111
(...)

Number of Leaves  :         13
Size of the tree :         25

=== Summary ===
Correctly Classified Instances       26532               95.7627 %
Incorrectly Classified Instances      1174                4.2373 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 8233  181   44    0 |    a = cluster0
  116 8541  181   92 |    b = cluster1
  177  234 7474    0 |    c = cluster2
    0  149    0 2284 |    d = cluster3
```

Tabla 16: Análisis de tasaciones - J48 ampliado. Prueba 2. CF y REP

La Tabla 16 muestra unos resultados acordes con las anteriores pruebas, el porcentaje de acierto de clasificación es muy bueno, teniendo además un árbol pequeño. Sin embargo, lo que puede parecer una situación ideal en realidad no es tal, dado que un vistazo al árbol generado por el clasificador con REP (Ilustración 19), revela una perspectiva que no puede apreciarse a simple vista por los resultados de tasa de acierto.

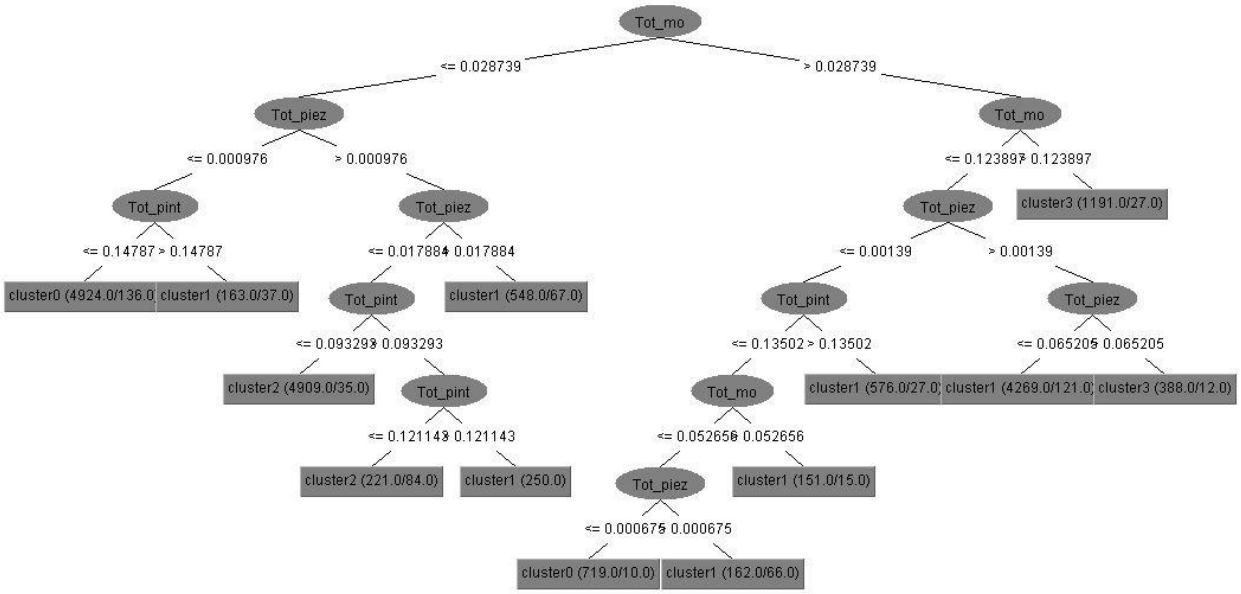


Ilustración 19: Análisis de tasaciones - J48 ampliado. Árbol generado por REP

Lo primero que salta la vista es el cambio de raíz, que pasa de ser Tot_piez a Tot_mo. Esta circunstancia se debe a la normalización, que altera los promedios y los máximos, haciendo que un atributo con menos dispersión como Tot_mo mejore en ganancia. Este hecho no alberga gran relevancia, pero un vistazo más exhaustivo también pone en evidencia que no aparecen por ninguna parte los nuevos atributos añadidos, lo que representa un problema.

La poda se ha llevado por delante las bifurcaciones dependientes de los atributos binarios debido a que no tienen una gran ganancia de información por sí solos. Los atributos de costes por tanto aglutinan toda la carga de decisión del modelo, lo que hace que este modelo sea similar al anterior, no aportando nada nuevo.

4.2.3. Modelo con variables binarias

Ya se ha visto en la sección anterior que una ampliación del *dataset* estudiado con nuevas variables no parecer ser la solución para obtener un modelo más completo. Conservar los atributos sobre los que se basó el *clustering* evidentemente crea un sesgo favorable a éstos, con lo que es necesario realizar la clasificación ignorando éstos atributos.

Una de las alternativas es realizar una clasificación atendiendo únicamente a los atributos binarios. Es más que probable que exista una relación entre las piezas cambiadas y la severidad del siniestro dado

que no en todas las circunstancias de accidente se reparan las mismas piezas. Por ejemplo, parece evidente que un accidente donde se tienen que cambiar piezas internas del motor tendrá una severidad superior a un percance en el cual sólo se pintan algunas piezas superficiales. Este tipo de conocimiento se puede extraer realizando una clasificación únicamente sobre los atributos binarios, que devolverá un conjunto de reglas de decisión.

A continuación se presenta una nueva batería de pruebas con los atributos binarios exclusivamente.

De nuevo empezando con la configuración por defecto de WEKA para J48 con CF 0.25 y minNumObj igual a 2. Se irán progresivamente afinando los parámetros.

En este caso no es necesario normalizar el *dataset*, los atributos sólo presentan valores de 0 o 1. Se aplica validación cruzada k=10 y aleatorización. Como ocurría en las pruebas anteriores, se ha optado por presentar la clasificación con los dos tipos de poda, CF y REP.

En la primera prueba, para CF, se han utilizado los parámetros por defecto de WEKA, con factor 0.25 y minNumObj 2. Con REP, el minNumObj también es 2 y el número de *folds* se establece en 3.

```
Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Instances:    27706
Attributes:   106
(...)

Number of Leaves  :         925
Size of the tree :         1849

=== Summary ===
Correctly Classified Instances       22632               81.6863 %
Incorrectly Classified Instances      5074               18.3137 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7861  317  275    5 |    a = cluster0
  751 6692 1067  420 |    b = cluster1
  628  694 6552   11 |    c = cluster2
   42  813   51 1527 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 5 -M 2
Instances:    27706
Attributes:   106
(...)

Number of Leaves  :         717
Size of the tree :         1433

=== Summary ===
Correctly Classified Instances       22568               81.4553 %
Incorrectly Classified Instances      5138               18.5447 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7867  329  261    1 |    a = cluster0
  747 6801 1013  369 |    b = cluster1
  636  786 6455    8 |    c = cluster2
   38  901   49 1445 |    d = cluster3
```

Tabla 17: Análisis de tasaciones - J48 binarios. Prueba 1. CF y REP

Un primer vistazo a la Tabla 17 revela una peor tasa de clasificación, sensiblemente más baja que en los casos anteriores. Esta circunstancia es completamente normal debido a que el *clustering* inicial no se hizo basado en estos atributos. De todas formas, un 81% no se considera un resultado demasiado malo en este caso.

Sin embargo, el tamaño del árbol es un dato que sí debe mejorarse, ambos tipos de poda generan árboles inmensos, que son difícilmente manejables o representables. Se realizarán más pruebas afinando los parámetros para ambos tipos de poda.

Para las siguientes pruebas se ha seguido la misma mecánica que en la obtención del primer modelo, afinando progresivamente los parámetros del algoritmo J48 para favorecer la post-poda. Para la alternativa con CF, se harán pruebas con factores de confianza 0.05, 0.01 y 0.001, aumentando también respectivamente el minNumObj a 50, 100 y 200. Para la poda REP se mantendrán también éstos minNumObj de 50, 100 y 200, con el número de *folds* en 3.

Las siguientes tablas (Tabla 18, Tabla 19, Tabla 20) reflejan las pruebas sobre el *dataset* de atributos binarios. Al final de la sección se muestra una comparativa (Tabla 21) de las pruebas realizadas.

```
Scheme:       weka.classifiers.trees.J48 -C 0.05 -M 50
Instances:    27706
Attributes:   106
(...)

Number of Leaves  :         100
Size of the tree :         199

=== Summary ===
Correctly Classified Instances       21568               77.846  %
Incorrectly Classified Instances      6138               22.154  %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7719  268  471    0 |    a = cluster0
  965 6301 1303  361 |    b = cluster1
  719  882 6276    8 |    c = cluster2
   68  995   98 1272 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 5 -M 50
Instances:    27706
Attributes:   106
(...)

Number of Leaves  :         90
Size of the tree :         179

=== Summary ===
Correctly Classified Instances       21327               76.9761 %
Incorrectly Classified Instances      6379               23.0239 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7631  282  545    0 |    a = cluster0
 1015 6243 1331  341 |    b = cluster1
  676  957 6242   10 |    c = cluster2
   68 1046  108 1211 |    d = cluster3
```

Tabla 18: Análisis de tasaciones - J48 binarios. Prueba 2. CF y REP

```
Scheme:       weka.classifiers.trees.J48 -C 0.01 -M 100
Instances:    27706
Attributes:   106
(...)

Number of Leaves  :         64
Size of the tree :         127

=== Summary ===
Correctly Classified Instances       21075               76.0666 %
Incorrectly Classified Instances      6631               23.9334 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7623  200  635    0 |    a = cluster0
 1197 6082 1318  333 |    b = cluster1
  781  934 6166    4 |    c = cluster2
   83 1046  100 1204 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 7 -M 100
Instances:    27706
Attributes:   106
(...)

Number of Leaves  :         61
Size of the tree :         121

=== Summary ===
Correctly Classified Instances       20489               73.9515 %
Incorrectly Classified Instances      7217               26.0485 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7410  279  769    0 |    a = cluster0
 1265 5752 1580  333 |    b = cluster1
  783  889 6206    7 |    c = cluster2
   72 1101  139 1121 |    d = cluster3
```

Tabla 19: Análisis de tasaciones - J48 binarios. Prueba 3. CF y REP

```
Scheme:       weka.classifiers.trees.J48 -C 0.001 -M 200
Instances:    27706
Attributes:   106
(...)

Number of Leaves  :         40
Size of the tree :         79

=== Summary ===
Correctly Classified Instances       19988               72.1432 %
Incorrectly Classified Instances      7718               27.8568 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7363  206  889    0 |    a = cluster0
 1621 5495 1538  276 |    b = cluster1
  817  860 6180   28 |    c = cluster2
  133 1244  106  950 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 5 -Q 5 -M 200
Instances:    27706
Attributes:   106
(...)
```

```
Number of Leaves  :         36
Size of the tree :         71

=== Summary ===
Correctly Classified Instances       19691               71.0712 %
Incorrectly Classified Instances      8015               28.9288 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7129  394  935    0 |    a = cluster0
 1355 5258 1952  365 |    b = cluster1
  769  805 6277   34 |    c = cluster2
  108 1063  235 1027 |    d = cluster3
```

Tabla 20: Análisis de tasaciones - J48 binarios. Prueba 4. CF y REP

Como se puede observar en la serie de pruebas, la penalización que implica la poda es mucho mayor sobre este conjunto de datos que sobre el anterior, aun utilizando los mismos parámetros J48 para las pruebas que en la sección 4.2.1. Debido al tamaño de árbol medio, mucho mayor en este caso, el cambio de parámetros afecta mucho más en términos de precisión, reducción del árbol y otros aspectos, dado que si se permite la poda, muchos más nodos se verán afectados por ella.

La Tabla 21 refleja precisamente esas diferencias. El aumento del minNumObj en ambos tipos de poda reduce drásticamente el tamaño del árbol (entre la primera y la segunda prueba ambos casos presentan una reducción cercana al 90%), pero la penalización en precisión también es mayor.

Sin duda, con el aumento de dimensionalidad, al pasar de apenas 3 atributos a 105, las cosas se "complican" para el clasificador, que debe ahora atender a muchas más variables.

Prueba	CF				REP			
	CF	minNumObj	Tamaño	Precisión	Folds	minNumObj	Tamaño	Precisión
1	0.25	2	1849	81,69%	3	2	1433	81,45%
2	0.05	50	199	77,85%	3	50	179	76,98%
3	0.01	100	127	76,07%	3	100	121	73,95%
4	0.001	200	79	72,14%	5	200	71	71,07%

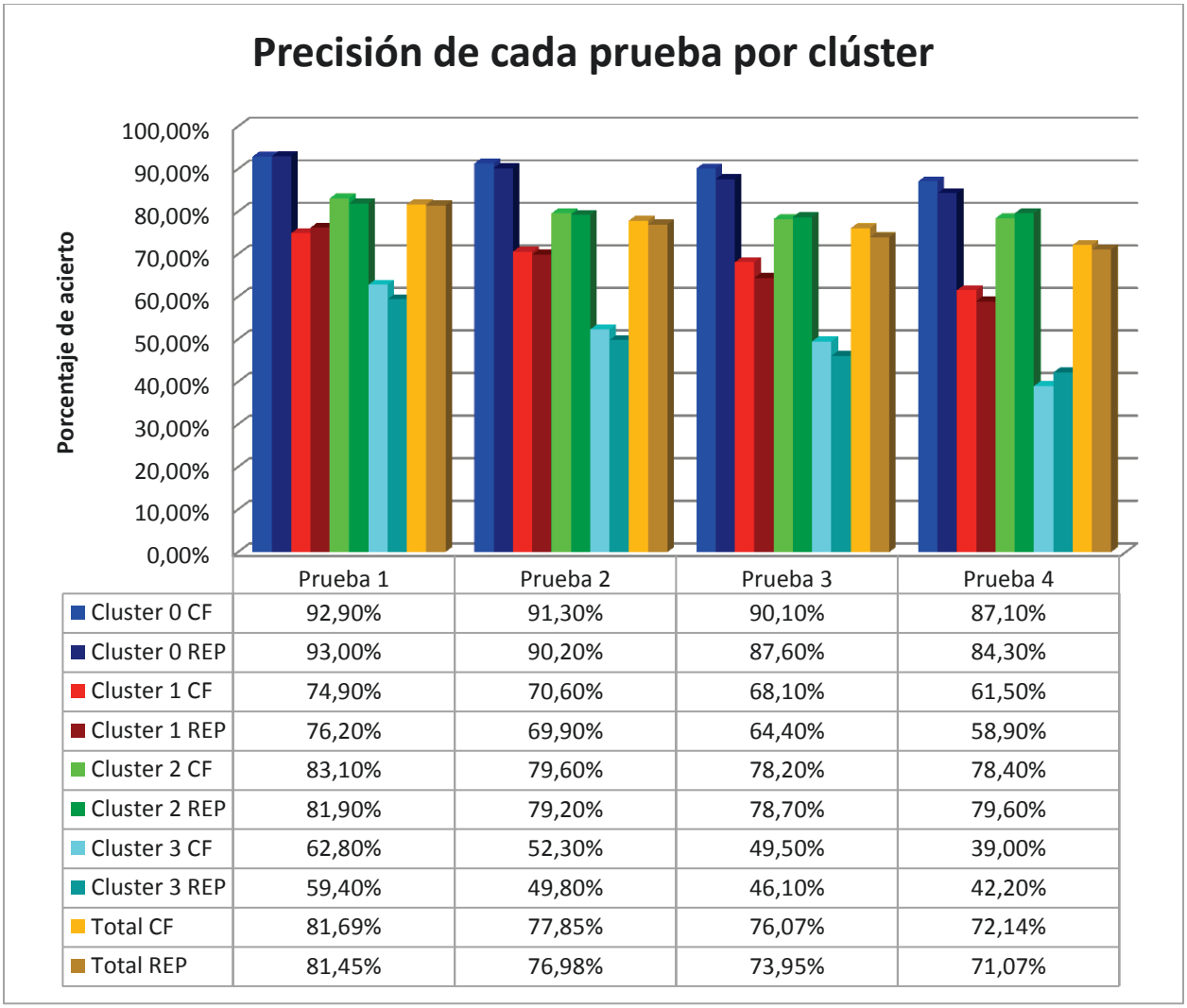
Tabla 21: Análisis de tasaciones – J48 binarios. Comparativa de pruebas

Si hubiera que decantarse por un modelo, el obtenido por CF en la prueba 4 es el que mejor porcentaje de acierto frente a tamaño de árbol mantiene, aunque aún existen ciertas características de los datos recogidos interesantes de analizar primero. En el ANEXO C: Salidas de WEKA puede consultarse la salida de WEKA completa del modelo elegido.

Un dato que llama la atención es la variación de precisión del clasificador dependiendo del clúster al que pertenezca la instancia. Según se observa, la tasa de acierto tiende a descender dentro de la misma prueba dependiendo del clúster que tenga que asignar el clasificador.

Como refleja la Gráfica 16, la precisión es diferente dependiendo del clúster al que pertenezca la instancia clasificada en cada momento. Cuando el clasificador procesa instancias pertenecientes a los clústeres 0 y 2 (grados 1 y 2), suele equivocarse poco (en ningún caso baja del 78% de tasa de acierto),

sin embargo, en los clústeres 1 y 3 (categorías de mayor severidad), la precisión desciende sobremanera, no superando el 50% en las tres últimas pruebas para el clúster 3 (39% en la última).



	Prueba 1	Prueba 2	Prueba 3	Prueba 4
Cluster 0 CF	92,90%	91,30%	90,10%	87,10%
Cluster 0 REP	93,00%	90,20%	87,60%	84,30%
Cluster 1 CF	74,90%	70,60%	68,10%	61,50%
Cluster 1 REP	76,20%	69,90%	64,40%	58,90%
Cluster 2 CF	83,10%	79,60%	78,20%	78,40%
Cluster 2 REP	81,90%	79,20%	78,70%	79,60%
Cluster 3 CF	62,80%	52,30%	49,50%	39,00%
Cluster 3 REP	59,40%	49,80%	46,10%	42,20%
Total CF	81,69%	77,85%	76,07%	72,14%
Total REP	81,45%	76,98%	73,95%	71,07%

Gráfica 16: Análisis de tasaciones – J48 binarios. Comparativa de precisión por clúster

En la comparativa clúster a clúster, se verifica que la poda CF consigue mejores porcentajes en prácticamente todas las situaciones, pero con un tamaño de árbol ligeramente superior.

Cabe añadir que si se observa la matriz de confusión de la última prueba, aquella con los peores resultados, ésta refleja un dato bastante interesante: cuando el clasificador falla en los clústeres 1 y 3, suele hacerlo asignando a la instancia un valor de severidad menor que el que le correspondería. En el caso del clúster 3 es evidente porque es el último clúster y sólo puede fallar a la baja, pero el caso del clúster 1 es especialmente curioso porque casi nunca falla asignándole un valor más alto del que le corresponde.

En la Tabla 22 se representa la matriz de confusión de la poda CF de la prueba 4, con los valores acertados para los clústeres 1 y 3 en azul y en rojo los fallos.

```
=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7363  206  889    0 |    a = cluster0
 1621 5495 1538  276 |    b = cluster1
  817  860 6180   28 |    c = cluster2
  133 1244  106  950 |    d = cluster3
```

Tabla 22: Análisis de tasaciones - J48 binarios. Matriz de confusión prueba 4 CF

Esta circunstancia permite afirmar que el modelo obtenido sin duda tiende a realizar valoraciones de severidad a la baja. Esta circunstancia se añade al ya poco deseable hecho de que el clasificador obtiene una baja tasa de acierto para los grados de mayor coste (considerados los más sensibles), haciendo de este un modelo no demasiado fiable, por lo que quizá sea conveniente continuar con la búsqueda.

Antes de optar por un modelo final, se probará con una última combinación de atributos que aún no se ha contemplado.

4.2.4. Modelo con variables binarias y número de piezas

Las pruebas anteriores han demostrado que es posible obtener un modelo de conocimiento a partir de un *dataset* que no incluya los atributos de costes totales, pero la prueba con los atributos binarios ha arrojado resultados no del todo satisfactorios.

Sin embargo, aún existen variables que no se han incluido en el análisis, los atributos Pos_int y Pos_mod. Éstos representan la cantidad de piezas totales cambiadas y reparadas en cada siniestro, no son una suma de los atributos binarios, sino que registran el total de piezas cambiadas real. Debido a esta circunstancia, es muy posible que puedan aportar algún tipo de ganancia sobre la decisión de clasificación, por lo que se incluirán en un nuevo conjunto de datos que será sometido a nuevas pruebas.

Por tanto, en la nueva baterías de pruebas se van a tener en cuenta los siguientes atributos:

- Los 105 atributos binarios de operaciones de pintura, reparación y sustitución.
- Los 2 atributos de conteo de piezas reparadas y sustituidas: Pos_int y Pos_mod.

En este caso tampoco es necesario normalizar el *dataset*, la conjunción de atributos numéricos con una misma unidad de medida y valores binarios 0 o 1 no plantea problemas. Como siempre, se aplica validación cruzada k=10 y aleatorización.

De nuevo se ha optado por realizar la clasificación con los dos tipos de poda, CF y REP.

La primera prueba siempre se ejecuta bajo los parámetros por defecto de WEKA, con un factor de confianza 0.25 para CF y un número de *folds* para REP de 3. Ambas alternativas comienzan con un minNumObj igual a 2 (ver Tabla 23).

```
Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Instances:    27706
Attributes:   108
(...)

Number of Leaves  :         892
Size of the tree :         1783

=== Summary ===
Correctly Classified Instances       23425               84.5485 %
Incorrectly Classified Instances      4281               15.4515 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7808  327  323    0 |    a = cluster0
  579 7139  848  364 |    b = cluster1
  506  644 6735    0 |    c = cluster2
    6  671   13 1743 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 5 -M 2
Instances:    27706
Attributes:   108
(...)

Number of Leaves  :         727
Size of the tree :         1453

=== Summary ===
Correctly Classified Instances       23384               84.4005 %
Incorrectly Classified Instances      4322               15.5995 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7745  363  350    0 |    a = cluster0
  576 7245  794  315 |    b = cluster1
  522  670 6693    0 |    c = cluster2
    6  712   14 1701 |    d = cluster3
```

Tabla 23: Análisis de tasaciones - J48 binarios y piezas. Prueba 1. CF y REP

A primera vista, los resultados con la primera prueba son ligeramente mejores que los conseguidos en la anterior experimentación, reduciéndose con ambos tipos de poda el tamaño de árbol ligeramente y mejorando la tasa de acierto en la clasificación.

Para las siguientes pruebas se va a seguir la misma mecánica que viene aplicándose en todo el análisis, consistente en 3 pruebas más que irán afinando progresivamente el algoritmo hasta conseguir resultados. Para la poda con CF, se harán experimentos con factores de confianza 0.05, 0.01 y 0.001, aumentando también respectivamente el minNumObj a 50, 100 y 200. Para la poda REP se mantendrán los mismos minNumObj salvo en la última prueba, que conviene dejar en 150. En la Tabla 27 se puede ver un resumen de todas las pruebas.

Los resultados de las pruebas están representados en la Tabla 24, Tabla 25 y Tabla 26:

```
Scheme:       weka.classifiers.trees.J48 -C 0.05 -M 50
Instances:    27706
Attributes:   108
(...)

Number of Leaves  :         88
Size of the tree :         175

=== Summary ===
Correctly Classified Instances       22925               82.7438 %
Incorrectly Classified Instances      4781               17.2562 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7778  384  296    0 |    a = cluster0
  690 7156  839  245 |    b = cluster1
  902  624 6359    0 |    c = cluster2
   13  777   11 1632 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 5 -M 50
Instances:    27706
Attributes:   108
(...)

Number of Leaves  :         76
Size of the tree :         151

=== Summary ===
Correctly Classified Instances       22679               81.8559 %
Incorrectly Classified Instances      5027               18.1441 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7653  392  413    0 |    a = cluster0
  725 6944  904  357 |    b = cluster1
  896  582 6407    0 |    c = cluster2
   13  730   15 1675 |    d = cluster3
```

Tabla 24: Análisis de tasaciones - J48 binarios y piezas. Prueba 2. CF y REP

```
Scheme:       weka.classifiers.trees.J48 -C 0.01 -M 100
Instances:    27706
Attributes:   108
(...)

Number of Leaves  :         53
Size of the tree :         105

=== Summary ===
Correctly Classified Instances       22242               80.2786 %
Incorrectly Classified Instances      5464               19.7214 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7414  689  355    0 |    a = cluster0
  671 7086  931  242 |    b = cluster1
 1125  566 6194    0 |    c = cluster2
   23  849   13 1548 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 7 -M 100
Instances:    27706
Attributes:   108
(...)

Number of Leaves  :         49
```

```
Size of the tree :          97

=== Summary ===
Correctly Classified Instances       21800               78.6833 %
Incorrectly Classified Instances      5906               21.3167 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7289  593  576    0 |    a = cluster0
  819 6739  936  436 |    b = cluster1
 1196  604 6085    0 |    c = cluster2
   24  711   11 1687 |    d = cluster3
```

Tabla 25: Análisis de tasaciones - J48 binarios y piezas. Prueba 3. CF y REP

```
Scheme:       weka.classifiers.trees.J48 -C 0.001 -M 200
Instances:    27706
Attributes:   108
(...)

Number of Leaves  :         28
Size of the tree :          55

=== Summary ===
Correctly Classified Instances       21346               77.0447 %
Incorrectly Classified Instances      6360               22.9553 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 6699  880  879    0 |    a = cluster0
  698 6922  967  343 |    b = cluster1
 1150  570 6165    0 |    c = cluster2
   22  835   16 1560 |    d = cluster3
```

```
Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 7 -M 150
Instances:    27706
Attributes:   108
(...)

Number of Leaves  :         31
Size of the tree :          61

=== Summary ===
Correctly Classified Instances       21369               77.1277 %
Incorrectly Classified Instances      6337               22.8723 %

=== Confusion Matrix ===
    a    b    c    d   <-- classified as
 7005  552  901    0 |    a = cluster0
  925 6710  926  369 |    b = cluster1
 1207  554 6124    0 |    c = cluster2
   23  863   17 1530 |    d = cluster3
```

Tabla 26: Análisis de tasaciones - J48 binarios y piezas. Prueba 4. CF y REP

En la prueba 4, la poda REP ha demostrado mejores resultados con un minNumObj de 150 en lugar de 200, es por ello que en la tabla anterior el parámetro figura con ese valor.

Tras la ejecución de la batería de pruebas, se puede comprobar que el resultado es mejor que con el anterior *dataset*, dado que los porcentajes globales de acierto en la clasificación han mejorado y, además, los árboles generados son de menor tamaño.

Echando un vistazo a la Tabla 27, en cada una de las pruebas tanto en CF como en REP los resultados son mejores que la sección anterior, destacando sobre todo la tasa de acierto, que en todas las pruebas es alrededor de un 5% más alta.

Prueba	CF				REP			
	CF	minNumObj	Tamaño	Precisión	Folds	minNumObj	Tamaño	Precisión
1	0.25	2	1783	84,55%	3	2	1453	84,40%
2	0.05	50	175	82,74%	3	50	151	81,85%
3	0.01	100	105	80,28%	3	100	97	78,68%
4	0.001	200	55	77,04%	3	150	61	77,13%

Tabla 27: Análisis de tasaciones – J48 binarios y piezas. Comparativa de pruebas

Atendiendo a esta comparativa, se ha considerado el modelo obtenido por CF en la prueba 4 como el mejor modelo, dado que obtiene un aceptable 77,04% de acierto con tan solo 55 nodos de tamaño de árbol. La salida completa de WEKA y el árbol generado se pueden consultar en el ANEXO C: Salidas de WEKA.

Sin duda, los resultados avalan la decisión de incluir los atributos Pos_int y Pos_mod en el conjunto de datos. Si se observa el árbol generado con más detalle (Ilustración 20), Pos_int se sitúa precisamente como nodo raíz del árbol, lo que indica que realiza un gran aporte de información para la decisión. El otro atributo, sin embargo, no parece tener tanta relevancia en el conjunto, dado que aparece muy cercano a las hojas (no aparece en la imagen pero si en las reglas).

Pueden además distinguirse ciertas decisiones dentro del modelo, como por ejemplo que los siniestros donde sea necesario sustituir el airbag (SUST_68101), el ventilador del motor (SUST_19201) o la suspensión trasera (SUST_42103), y acumulen más de 12 piezas sustituidas (Pos_int>12), corresponderán al grado de mayor severidad (grado 4, clúster 3).

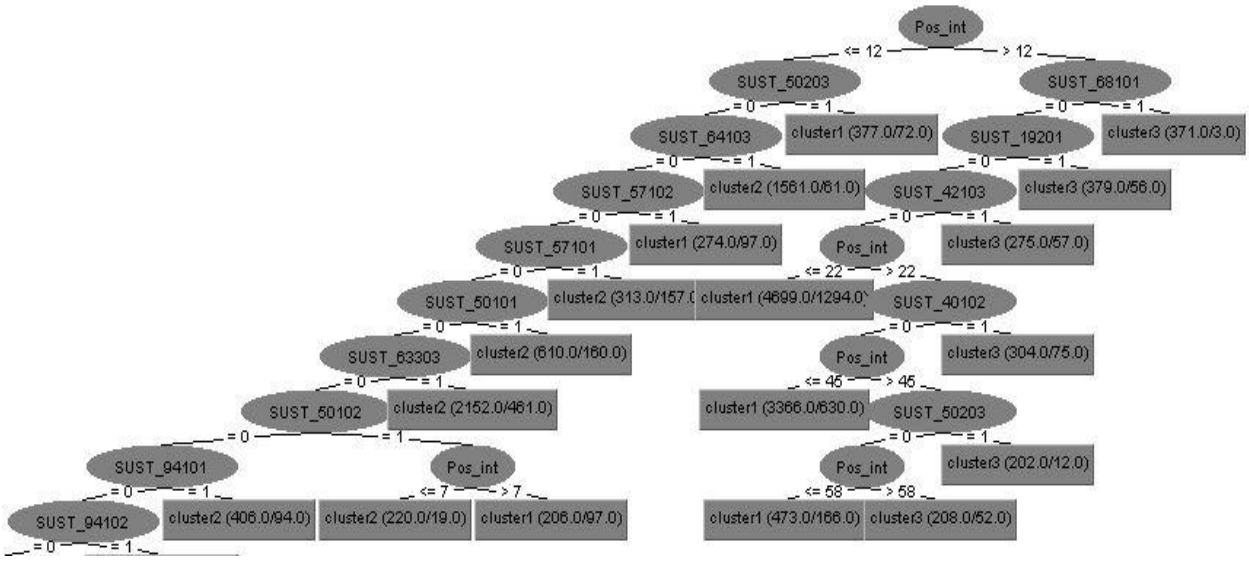


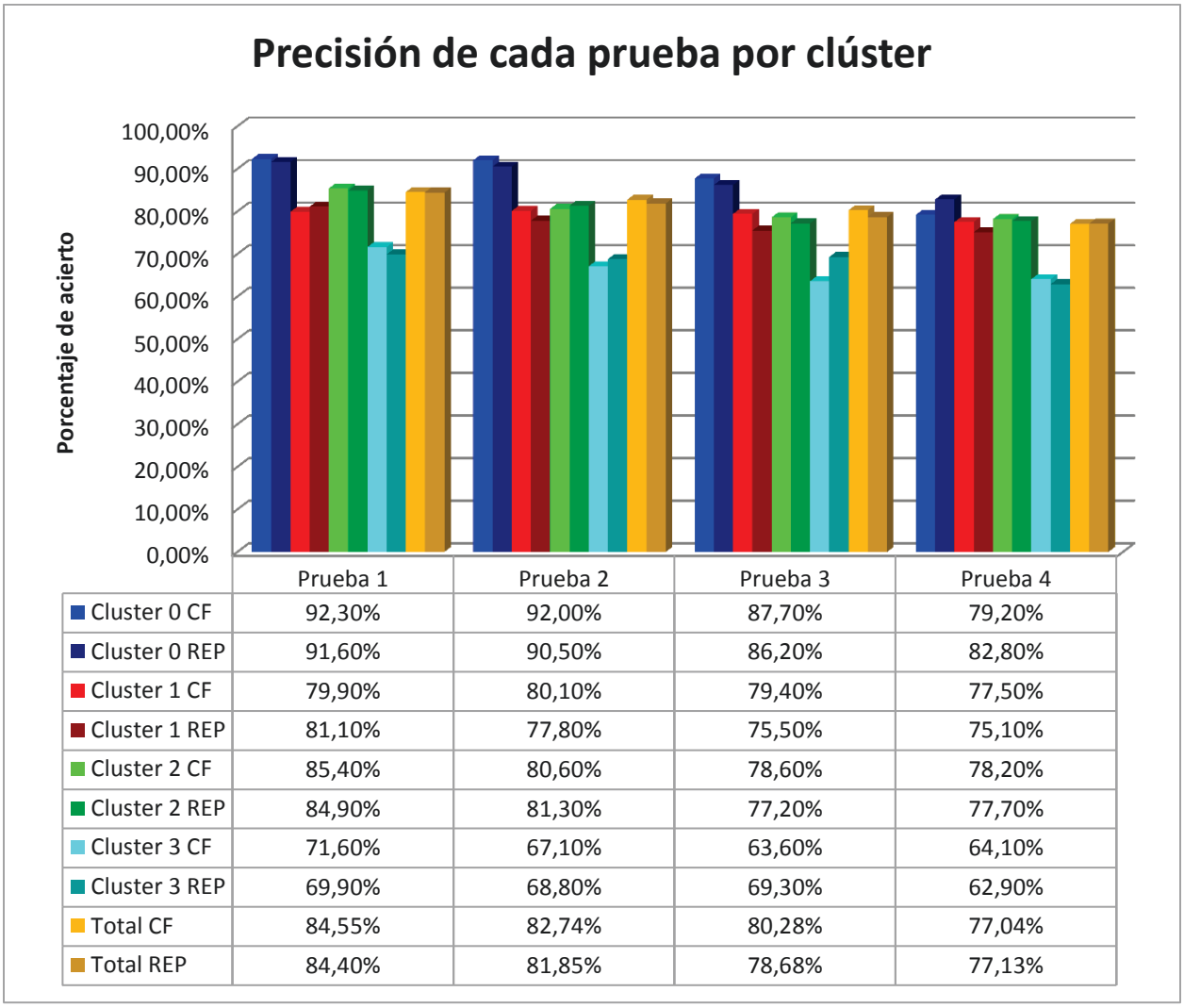
Ilustración 20: Análisis de tasaciones - J48 binarios y piezas. Árbol del mejor modelo (detalle).

Pese a la alta dimensionalidad del conjunto, se ha obtenido un modelo no demasiado intrincado que presenta un porcentaje aceptable, considerando que el 75% de acierto es el umbral mínimo a superar. Éstas son las reglas de decisión asociadas al modelo (Tabla 28):

```
J48 pruned tree
------------------
Pos_int <= 12
| SUST_50203 = 0
| | SUST_64103 = 0
| | | SUST_57102 = 0
| | | | SUST_57101 = 0
| | | | | SUST_50101 = 0
| | | | | | SUST_63303 = 0
| | | | | | | SUST_50102 = 0
| | | | | | | | SUST_94101 = 0
| | | | | | | | | SUST_94102 = 0
| | | | | | | | | | SUST_66501 = 0
| | | | | | | | | | | SUST_63203 = 0
| | | | | | | | | | | | SUST_94201 = 0
| | | | | | | | | | | | | Pos_mod <= 0
| | | | | | | | | | | | | | SUST_66202 = 0
| | | | | | | | | | | | | | | PT_53101 = 0
| | | | | | | | | | | | | | | | SUST_66201 = 0
| | | | | | | | | | | | | | | | | Pos_int <= 5: cluster2 (1128.0/421.0)
| | | | | | | | | | | | | | | | | Pos_int > 5: cluster0 (285.0/167.0)
| | | | | | | | | | | | | | | | SUST_66201 = 1: cluster0 (208.0/53.0)
| | | | | | | | | | | | | | | PT_53101 = 1: cluster0 (279.0/73.0)
| | | | | | | | | | | | | | SUST_66202 = 1: cluster0 (458.0/113.0)
| | | | | | | | | | | | | Pos_mod > 0: cluster0 (7363.0/1436.0)
| | | | | | | | | | | | SUST_94201 = 1: cluster2 (295.0/129.0)
| | | | | | | | | | | SUST_63203 = 1: cluster2 (703.0/170.0)
| | | | | | | | | | SUST_66501 = 1: cluster2 (298.0/44.0)
| | | | | | | | | SUST_94102 = 1: cluster2 (293.0/64.0)
| | | | | | | | SUST_94101 = 1: cluster2 (406.0/94.0)
| | | | | | | SUST_50102 = 1
| | | | | | | | Pos_int <= 7: cluster2 (220.0/19.0)
| | | | | | | | Pos_int > 7: cluster1 (206.0/97.0)
| | | | | | SUST_63303 = 1: cluster2 (2152.0/461.0)
| | | | | SUST_50101 = 1: cluster2 (610.0/160.0)
| | | | SUST_57101 = 1: cluster2 (313.0/157.0)
| | | SUST_57102 = 1: cluster1 (274.0/97.0)
| | SUST_64103 = 1: cluster2 (1561.0/61.0)
| SUST_50203 = 1: cluster1 (377.0/72.0)
Pos_int > 12
| SUST_68101 = 0
| | SUST_19201 = 0
| | | SUST_42103 = 0
| | | | Pos_int <= 22: cluster1 (4699.0/1294.0)
| | | | Pos_int > 22
| | | | | SUST_40102 = 0
| | | | | | Pos_int <= 45: cluster1 (3366.0/630.0)
| | | | | | Pos_int > 45
| | | | | | | SUST_50203 = 0
| | | | | | | | Pos_int <= 58: cluster1 (473.0/166.0)
| | | | | | | | Pos_int > 58: cluster3 (208.0/52.0)
| | | | | | | SUST_50203 = 1: cluster3 (202.0/12.0)
| | | | | | SUST_40102 = 1: cluster3 (304.0/75.0)
| | | SUST_42103 = 1: cluster3 (275.0/57.0)
| | SUST_19201 = 1: cluster3 (379.0/56.0)
| SUST_68101 = 1: cluster3 (371.0/3.0)
```

Tabla 28: Análisis de tasaciones – J48 binarios y piezas. Reglas de decisión asociadas al mejor modelo

No sólo los porcentajes generales son mejores que en el conjunto con sólo atributos binarios, sino que la precisión clúster a clúster también es mejor, como se puede apreciar en la Gráfica 17. El nuevo modelo tiene un ligera pérdida de porcentaje de acierto sobre los clústeres de menor coste (0 y 2), pero mejora en gran medida en los clústeres de mayor coste (1 y 3), equilibrando de esta manera los porcentajes. El peor valor sigue situándose en la prueba 4 sobre el clúster 3, pero en este caso consigue una precisión un 20% mayor que en el *dataset* anterior, con un 63% de acierto frente al 39% que reflejaba la anterior gráfica (Gráfica 16).



	Prueba 1	Prueba 2	Prueba 3	Prueba 4
Cluster 0 CF	92,30%	92,00%	87,70%	79,20%
Cluster 0 REP	91,60%	90,50%	86,20%	82,80%
Cluster 1 CF	79,90%	80,10%	79,40%	77,50%
Cluster 1 REP	81,10%	77,80%	75,50%	75,10%
Cluster 2 CF	85,40%	80,60%	78,60%	78,20%
Cluster 2 REP	84,90%	81,30%	77,20%	77,70%
Cluster 3 CF	71,60%	67,10%	63,60%	64,10%
Cluster 3 REP	69,90%	68,80%	69,30%	62,90%
Total CF	84,55%	82,74%	80,28%	77,04%
Total REP	84,40%	81,85%	78,68%	77,13%

Gráfica 17: Análisis de tasaciones – J48 binarios y piezas. Comparativa de precisión por clúster

Sin duda, este modelo es sensiblemente mejor que el obtenido únicamente con atributos binarios, por lo que se escogerá este como mejor modelo no basado en costes.

4.2.5. Elección del mejor modelo

La experimentación anterior ha arrojado dos modelos de decisión diferentes perfectamente funcionales, por lo que se plantea una encrucijada a la hora de escoger qué modelo es mejor.

El modelo inferido a partir de los atributos de coste presenta un excelente porcentaje de acierto en la clasificación (96%) y un tamaño de árbol muy reducido con apenas 23 nodos, por lo que es el más fiable, manejable y ligero de los obtenidos. Sin embargo, se considera fácil de burlar en caso de fraude porque se basa en datos no tangibles como son los costes, con lo que a un estafador le basta con falsificar los costes de una factura para conseguir el engaño. Además, el hecho de basarse en precios de piezas y mano de obra hace que no sea transportable a diferentes entornos donde se utilicen piezas de otros fabricantes, aparte de ser vulnerable a fluctuaciones del mercado y a la inflación anual.

El modelo inferido a partir de los atributos binarios y conteo de piezas ha demostrado ser suficientemente fiable como para ser tenido en consideración, con un 77% de tasa de acierto y 55 nodos. Pese a ser menos fiable que el primero mencionado, su principal ventaja radica en que no depende de los costes, con lo que es válido para un periodo de tiempo mucho más largo y no es sensible a las fluctuaciones de precios que si afectan al primero. Al tratarse de un modelo basado en piezas físicas, es posible contrastar las reparaciones realizadas sobre el vehículo y verificar si existen tasaciones fraudulentas.

Dado que ambos modelos presentan ventajas a tener en cuenta, lo más conveniente será elegir ambos y considerarlos complementarios.

4.3.Auditoría de daños

Según se menciona en la sección de Detección de fraude (2.2.2), un 34% de los casos de estafa detectados corresponden a la ocultación de daños preexistentes, es decir, aprovechando que se da parte de un siniestro, se incluyen pequeños desperfectos ocurridos en otros percances para que el seguro los incluya en la reparación y así ahorrarse la penalización en la póliza. Éste sin duda uno de los casos más habituales de fraude perpetrado por el propio asegurado, y que en la mayoría de las ocasiones es difícil de detectar.

Una alternativa interesante sería desarrollar un mecanismo que permita contrastar las características de un siniestro y detectar qué desperfectos no encajan con la mecánica del accidente, lo que permitiría evaluar si dichos daños pudieran haber sido ocasionados previamente.

Este proyecto propone estudiar y diseñar un modelo que permita agrupar los siniestros por tipología de impacto (Ilustración 21) y poder determinar que piezas suelen ser las más habitualmente reparadas, sustituidas o pintadas dentro de los tipos resultantes. De esta manera, si se encuentran desperfectos que no se corresponden con la tipología del siniestro, se podría catalogar dicha instancia como sospechosa de fraude.

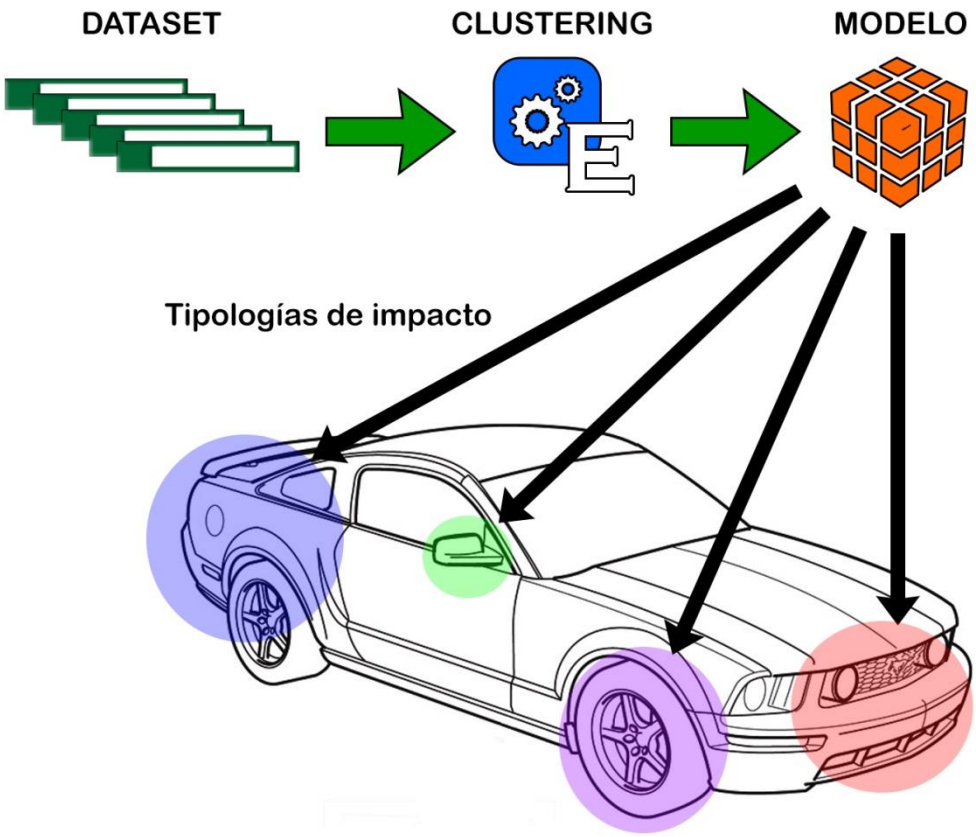


Ilustración 21: Análisis de tipologías de impacto

4.3.1. Modelado mediante *clustering*

Instintivamente, la primera opción para realizar una distinción dentro de un conjunto de datos es la de hacer *clustering* sobre los mismos. Como su propia definición indica (sección 2.4.2): "El *clustering* radica en agrupar diferentes objetos de tal manera que aquellos que pertenecen a un grupo (llamado clúster) se asemejen más entre sí que a aquellos presentes en otros grupos".

En el análisis de severidad, el fichero ya contaba con una clasificación realizada *a priori*, con lo que solamente era necesario proceder a los experimentos de clasificación y obtener resultados. En este caso, no se tiene conocimiento previo de la tendencia del *dataset*, por lo que será necesario realizar un agrupamiento sobre el conjunto de datos y estudiar las regiones resultantes.

Para el *clustering* se ha optado por utilizar de nuevo el algoritmo EM (*Expectation Maximization*) frente a K-medias, debido a que el primero garantiza convergencia y un mejor comportamiento si se establece un número fijo de clústeres [31]. Dado que no se conocen de primeras los tipos de impacto que pueden resultar del análisis, se irá progresivamente probando con diferente cantidad de clústeres para intentar obtener un resultado satisfactorio.

En la siguiente batería de pruebas, se ejecutará el algoritmo EM teniendo en cuenta los siguientes atributos:

- Los 105 atributos binarios de operaciones de pintura, reparación y sustitución.

El resto de atributos de coste y número de piezas se mantendrán en el *dataset* con el fin de poder ser observados *a posteriori,* pero serán ignorados durante las pruebas para evitar influir en el agrupamiento. El *clustering* se realizará exclusivamente con los atributos binarios.

Para la primera prueba, se establecerá el número de clústeres en 5, y se mantendrán los parámetros estándar que establece WEKA.

```
Scheme:         weka.clusterers.EM -I 100 -N -1 -X 10 -max 5 -ll-cv 1.0E-6 -ll-iter
1.0E-6 -M 1.0E-6 -K 10 -num-slots 2 -S 11
Instances:    27706
Attributes:   111
(…)

EM
==
Number of clusters selected by cross validation: 5
Number of iterations performed: 100

Clustered Instances
0      11576 ( 42%)
1       4445 ( 16%)
2       3361 ( 12%)
3       3053 ( 11%)
4       5271 ( 19%)

Log likelihood: -18.2359
```

Tabla 29: Auditoría de daños - EM. Prueba 1

Como se puede apreciar en la Tabla 29, el algoritmo ha clasificado todo el conjunto de datos en únicamente 5 clústeres, con una gran mayoría de instancias pertenecientes al primer clúster, y porcentajes mucho más pequeños asociados al resto de grupos. Esto, en teoría, no debería ser tan acusado, sin embargo, un análisis más exhaustivo de las instancias puede revelar más información.

Para determinar la zona de impacto y tipología del siniestro que cada clúster representa, se van a tener en cuenta aquellos atributos que superan un determinado "porcentaje de positivos" dentro de cada clúster. Este porcentaje corresponde al tanto por ciento de instancias en las cuales un determinado atributo presenta valor positivo dentro del total de instancias de un clúster. Esto es, en un clúster de 1000 instancias, por ejemplo, un porcentaje del 40% significará que en 400 instancias el atributo presenta valor 1, y en el resto, 0.

Los atributos con los porcentajes de positivos más altos son los más comunes dentro de un clúster en cuestión y por tanto "caracterizan" al mismo.

5 atributos más comunes			
Clúster	Código	Descripción	Porcentaje*
Clúster 0	PT_63303	Pintura paragolpes trasero central	47,03%
	SUST_63303	Sustitución paragolpes trasero central	23,70%
	PT_63203	Pintura paragolpes delantero central	21,90%
	REP_63303	Reparación paragolpes trasero central	21,73%
	SUST_66953	Sustitución Matrícula	17,74%
Clúster 1	SUST_66202	Sustitución molduras derecha	77,41%
	PT_57102	Pintura puerta derecha	70,69%
	PT_53102	Pintura aleta trasera derecha	61,62%
	PT_50102	Pintura aleta delantera derecha	50,84%
	REP_53102	Reparación aleta trasera derecha	50,84%
Clúster 2	SUST_63203	Sustitución paragolpes delantero central	92,17%
	PT_63203	Pintura paragolpes delantero central	91,52%
	SUST_50203	Sustitución componentes de la coraza central	72,92%
	PT_55103	Pintura capó central	69,18%
	SUST_94101	Sustitución faro delantero izquierdo	63,40%
Clúster 3	PT_53102	Pintura aleta trasera derecha	86,83%
	PT_53101	Pintura aleta trasera izquierda	84,97%
	PT_63303	Pintura paragolpes trasero central	81,95%
	PT_57102	Pintura puerta derecha	80,84%
	PT_63203	Pintura paragolpes delantero central	77,89%
Clúster 4	PT_57101	Pintura puerta izquierda	59,72%
	SUST_66201	Sustitución molduras izquierda	59,65%
	PT_50101	Pintura aleta delantera izquierda	55,95%
	PT_53101	Pintura aleta trasera izquierda	39,88%
	REP_57101	Reparación puerta izquierda	38,91%
***El porcentaje se refiere a la tasa de positivos del atributo en el total de instancias del clúster**			

Tabla 30: Auditoría de daños - Atributos más comunes con 5 clústeres

La Tabla 30 muestra una comparativa de los atributos más comunes por clúster. En ella, empieza a distinguirse una separación entre atributos correspondientes a diferentes zonas del vehículo. Por ejemplo, los clústeres 1 y 4 parece que tienden a contener únicamente instancias con reparaciones localizadas en las partes derecha e izquierda del vehículo, respectivamente, mientras que el clúster 2 parece aglutinar desperfectos correspondientes a la zona delantera. También es interesante observar como el clúster 3, pese a situarse en todas las zonas, tiende a aglutinar sólo daños correspondientes a pintura, lo que también debe ser tenido en cuenta en el modelo final.

Pese a ser prometedor, el resultado no se considera suficiente, dado que existen aún demasiadas piezas desordenadas entre clústeres. Aparte, la función de verosimilitud (*log likelihood* -18.2359) no se considera suficientemente buena. Uno de los objetivos del *clustering* con EM consiste en maximizar ésta función, acercando su valor a 0 en la medida de lo posible. Un valor más próximo a 0 indica una mejor adaptación del modelo al conjunto de datos.

Se ejecutarán 2 pruebas más con números de clúster 10 y 15, respectivamente, con el fin de demostrar que la función de verosimilitud se maximiza aumentando el número de clúster.

En la Tabla 31 y Tabla 32 se muestran los resultados de dichas pruebas:

```
Scheme:         weka.clusterers.EM -I 100 -N -1 -X 10 -max 10 -ll-cv 1.0E-6 -ll-iter
1.0E-6 -M 1.0E-6 -K 10 -num-slots 2 -S 100
Instances:    27706
Attributes:   111
(…)

EM
==
Number of clusters selected by cross validation: 10
Number of iterations performed: 29

Clustered Instances
0       1789 (  6%)
1       1347 (  5%)
2       1972 (  7%)
3       4221 ( 15%)
4       3046 ( 11%)
5        925 (  3%)
6       1846 (  7%)
7       4226 ( 15%)
8       3843 ( 14%)
9       4491 ( 16%)

Log likelihood: -16.52118
```

Tabla 31: Auditoría de daños - EM. Prueba 2

```
Scheme:         weka.clusterers.EM -I 100 -N -1 -X 10 -max 15 -ll-cv 1.0E-6 -ll-iter
1.0E-6 -M 1.0E-6 -K 10 -num-slots 2 -S 100
Instances:    27706
Attributes:   111
(…)

EM
==
Number of clusters selected by cross validation: 15
Number of iterations performed: 45

Clustered Instances
0       3266 ( 12%)
1       1082 (  4%)
2       2385 (  9%)
3       1487 (  5%)
4       1751 (  6%)
5       1247 (  5%)
6       2734 ( 10%)
7       1577 (  6%)
8       4085 ( 15%)
```

```
9         727 (  3%)
10       1732 (  6%)
11       3495 ( 13%)
12        682 (  2%)
13        815 (  3%)
14        641 (  2%)

Log likelihood: -15.75907
```

Tabla 32: Auditoría de daños - EM. Prueba 3

Los resultados de estas pruebas son bastante reveladores. En primer lugar, la distribución de porcentajes queda más homogénea que en la primera prueba, lo que puede significar que anteriormente, instancias con poca similitud podían estar englobadas dentro de un mismo clúster, y al forzar la creación de más grupos en el algoritmo de agrupamiento, las instancias se hayan repartido entre otros clústeres ahora. También, el número de iteraciones necesarias del algoritmo es menor, es decir, converge antes. Por último, la función de verosimilitud se ve reducida con el aumento de número de clústeres, lo que significa que el modelo ajusta mejor.

En la Tabla 33 se visualizan los resultados de función de verosimilitud de las 3 pruebas:

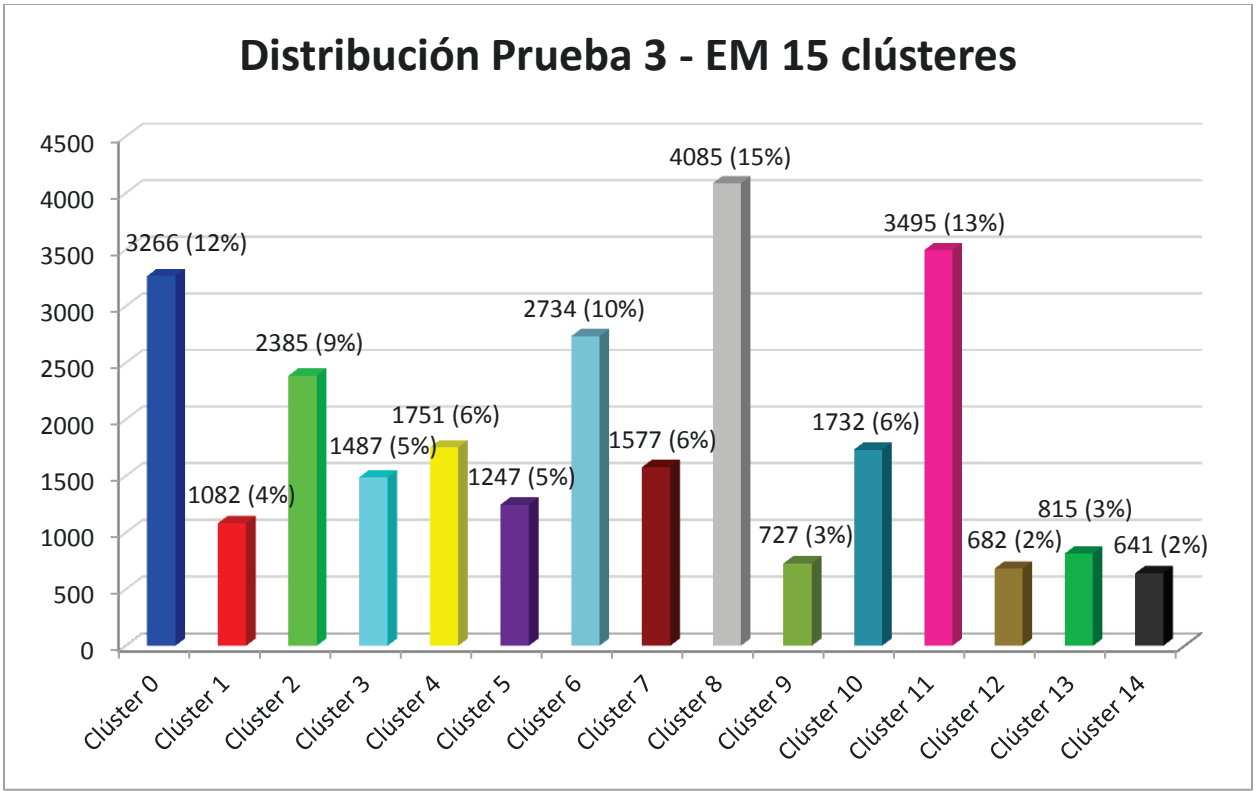
Prueba	Número clústeres	Iteraciones	Verosimilitud (*Likelihood*)
1	5	100	-18.2359
2	10	29	-16.52118
3	15	45	-15.75907

Tabla 33: Auditoría de daños - EM. Comparativa de pruebas

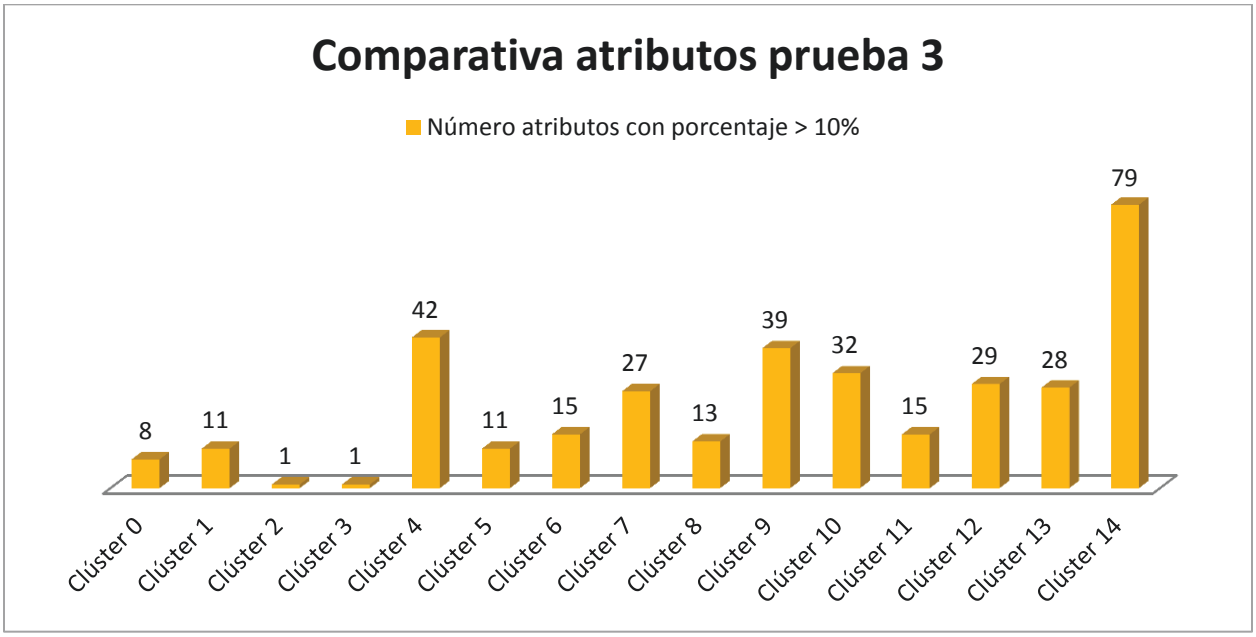
Sin embargo, las pruebas realizadas ofrecen también malos resultados en la separación de grupos, se generan clústeres con demasiados atributos que presentan un gran porcentaje de positivos, lo que indica que el modelo no se ajusta lo suficiente en algunos grupos como para poder establecer un modelo de garantías. Idealmente, debería poder establecerse una subdivisión de atributos dentro de cada clúster que sea claramente disjunta, o al menos más clara que la que se ha obtenido con las pruebas realizadas.

En cuanto a la prueba 3 con 15 clústeres, que obtiene los mejores resultados, no es posible plasmar en este documento las tablas de asignación de atributos a clústeres debido al desmesurado tamaño de éstas, por lo que se plantean otros métodos de visualización.

Estas dos gráficas (Gráfica 18, Gráfica 19) representan tanto la distribución por clúster como el número de atributos individuales que superan el 10% de porcentaje de positivos. Los atributos que superan el 10% de porcentaje de positivos se consideran los más comunes dentro de un clúster, y es una manera de determinar la zona de impacto que representará dicho clúster.



Gráfica 18: Auditoría de daños - EM. Distribución prueba 3



Gráfica 19: Auditoría de daños - EM. Comparativa atributos positivos prueba 3

Sin duda, de estas dos comparativas sobre todo llama la atención la gran cantidad de atributos que superan el 10% de porcentaje de positivos en los clústeres con relativamente pocas instancias (clústeres 4, 9 y 14). Los 3 clústeres señalados superan los 35 atributos con más de 10% de positivos, y sin embargo, entre todos no suman ni el 12% de las instancias. Este subconjunto de datos es residual, pero

parece suficiente como para enturbiar el modelo y restarle fiabilidad. La circunstancia de que un clúster registre tantos atributos con alto porcentaje de positivos se considera negativa, no puede establecerse una zona de impacto claramente definida si en ella se presentan todo tipo de reparaciones diferentes. Para conseguir buenos resultados, debería presentarse una distribución en la cual los atributos conformen grupos disjuntos entre clústeres, o al menos que guarden similitudes en zona del vehículo e intensidad.

Parece evidente que el algoritmo no es capaz de converger hacia un resultado óptimo con el número de clústeres planteado. Anteriormente se pudo constatar que aumentando el número de clústeres se podía conseguir una mejor verosimilitud, pero llegados a este punto, forzar a que el algoritmo incluya más clústeres no parece una buena solución, dado que el número de clústeres ya es susceptiblemente grande.

Según se expone en el estudio publicado en 2012 por Adebisi et al. [31], EM presenta problemas de optimalidad frente a conjuntos con alta dimensionalidad, pese a que garantiza convergencia en la inmensa mayoría de los casos. Con el *dataset* actual, quizá 27.000 instancias no sean suficientes para que el algoritmo pueda converger hacia un resultado óptimo.

Por suerte, se cuenta con un fichero mucho más grande de instancias sobre el que realizar la experimentación.

4.3.2. Ampliación del dataset

Con el fin de conseguir mejores resultados, se va a proceder a realizar una batería de pruebas similar a la de la sección anterior sobre un conjunto de datos de 151.826 instancias.

De nuevo, se ejecutará el algoritmo EM teniendo en cuenta exclusivamente los 105 atributos binarios que contienen las operaciones de pintura, reparación y sustitución. El resto de atributos seguirán manteniéndose ignorados pero presentes para realizar observaciones posteriores en la evaluación de resultados.

En esta ocasión, dado que se ha demostrado que con un número de clústeres por encima de 10 se obtienen mejores resultados, directamente se va a proceder a realizar pruebas sobre el *dataset* con 10, 12 y 15 clústeres.

Los parámetros del algoritmo en WEKA son los que aparecen por defecto, sólo se han variado los números de clúster en cada una de las pruebas.

En las tablas (Tabla 34, Tabla 35 y Tabla 36) se pueden observar los resultados de las pruebas:

```
Scheme:         weka.clusterers.EM -I 100 -N -1 -X 10 -max 10 -ll-cv 1.0E-6 -ll-iter
1.0E-6 -M 1.0E-6 -K 10 -num-slots 2 -S 11
Instances:    151826
Attributes:   111
(…)

EM
==
Number of clusters selected by cross validation: 10
Number of iterations performed: 50

Clustered Instances
 0       25571 ( 17%)
 1       10192 (  7%)
 2       12109 (  8%)
 3        6005 (  4%)
 4       10355 (  7%)
 5       30894 ( 20%)
 6       13325 (  9%)
 7        7246 (  5%)
 8       24679 ( 16%)
 9       11450 (  8%)

Log likelihood: -9.78503
```

Tabla 34: Auditoría de daños - EM ampliado. Prueba 1

```
Scheme:         weka.clusterers.EM -I 100 -N -1 -X 10 -max 12 -ll-cv 1.0E-6 -ll-iter
1.0E-6 -M 1.0E-6 -K 10 -num-slots 2 -S 11
Instances:    151826
Attributes:   111
(…)

EM
==
Number of clusters selected by cross validation: 12
Number of iterations performed: 62

Clustered Instances
 0       12036 (  8%)
 1       10111 (  7%)
 2       12825 (  8%)
 3       28267 ( 19%)
 4        7507 (  5%)
 5       27315 ( 18%)
 6        4018 (  3%)
 7        2628 (  2%)
 8       16708 ( 11%)
 9        7312 (  5%)
10        9956 (  7%)
11       13143 (  9%)

Log likelihood: -9.69458
```

Tabla 35: Auditoría de daños - EM ampliado. Prueba 2

```
Scheme:         weka.clusterers.EM -I 100 -N -1 -X 10 -max 15 -ll-cv 1.0E-6 -ll-iter
1.0E-6 -M 1.0E-6 -K 10 -num-slots 2 -S 23
Instances:    151826
Attributes:   111

EM
==
Number of clusters selected by cross validation: 15
Number of iterations performed: 38

Clustered Instances
 0       10023 (  7%)
 1        6238 (  4%)
 2        5726 (  4%)
 3        7433 (  5%)
 4        4421 (  3%)
 5        9226 (  6%)
 6       12594 (  8%)
 7       23075 ( 15%)
 8        6057 (  4%)
 9        4213 (  3%)
10       28936 ( 19%)
11       18304 ( 12%)
12        8245 (  5%)
13        4508 (  3%)
14        2827 (  2%)

Log likelihood: -9.41042
```

Tabla 36: Auditoría de daños - EM ampliado. Prueba 3

Atendiendo a los resultados de las pruebas, la función de verosimilitud de todas ellas es sustancialmente mejor (en todas mayor de -10) que en la sección anterior, lo que refleja que la inclusión de un *dataset* mayor ha sido un acierto.

Las distribuciones de instancias por cada clúster parecen mantener porcentajes similares a las realizadas con el conjunto de datos menor, con la única variación del orden, que por otra parte es natural dado que se han sido aleatorizadas.

La Tabla 37 establece la comparativa en las tres pruebas, para una mejor visualización en conjunto de los resultados:

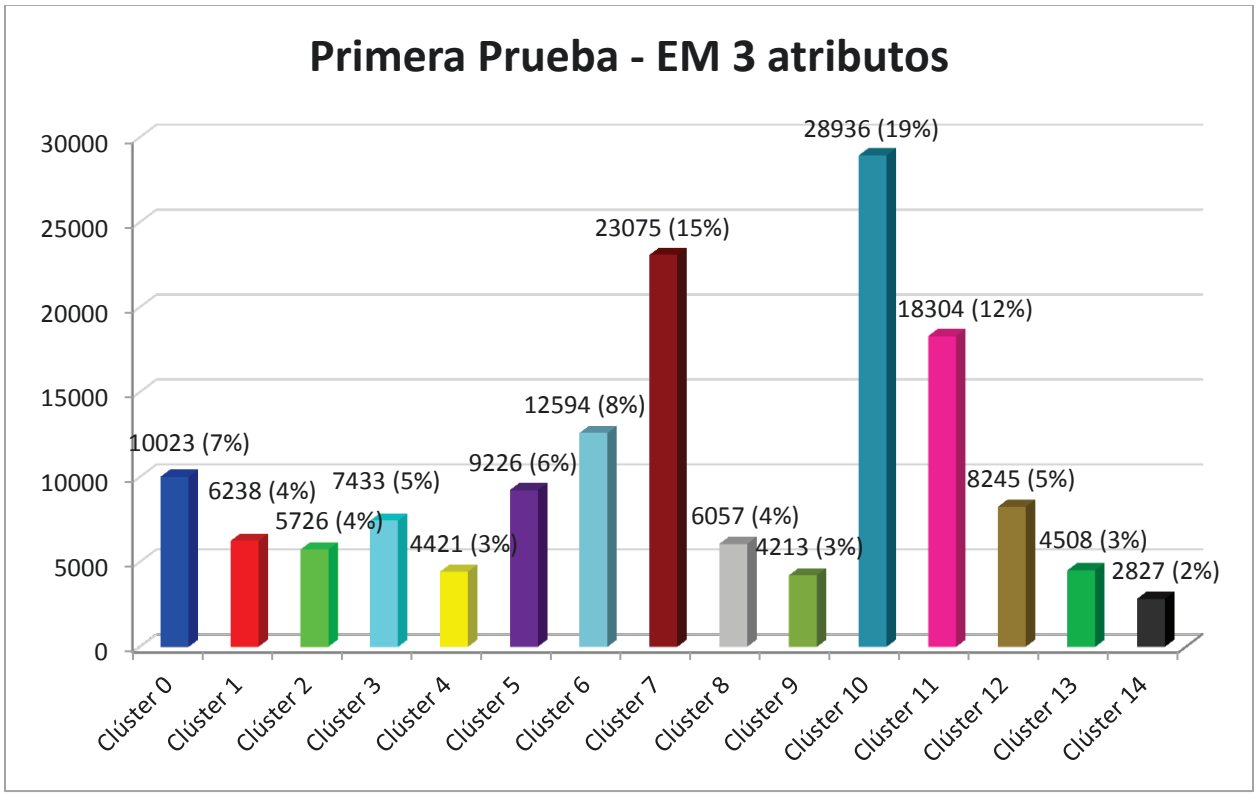
Prueba	Número clústeres	Iteraciones	Verosimilitud (*Likelihood*)
1	10	50	-9.78503
2	12	62	-9.69458
3	15	38	-9.41042

Tabla 37: Auditoría de daños - EM ampliado. Comparativa de pruebas

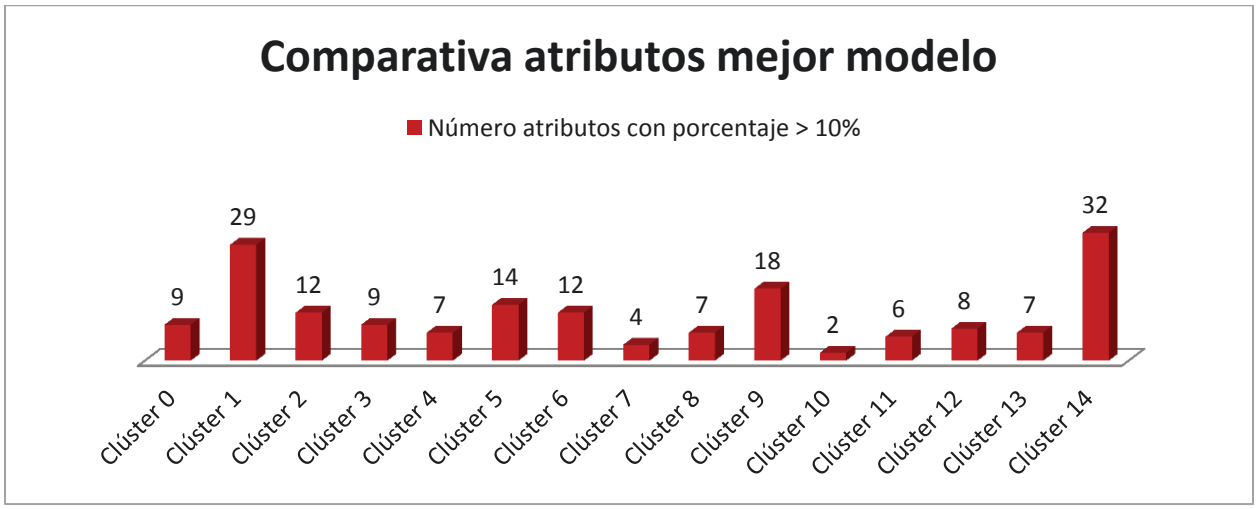
Según los resultados de verosimilitud e iteraciones, sin duda el modelo con 15 clústeres es el mejor. Resulta conveniente llegado este punto realizar un estudio pormenorizado de las características de este modelo para distinguir las zonas y tipologías de impacto presentes.

4.3.3. Elección del mejor modelo

Una vez decidido que el modelo de 15 clústeres sobre el *dataset* de 150.000 instancias es el mejor, se va a proceder a analizar las características del mismo. A continuación se presentan diversas gráficas y diagramas que detallan las tipologías de impacto trazadas por el *clustering* y la diferente intensidad de las mismas, si existe.



Gráfica 20: Auditoría de daños - EM ampliado. Distribución mejor modelo

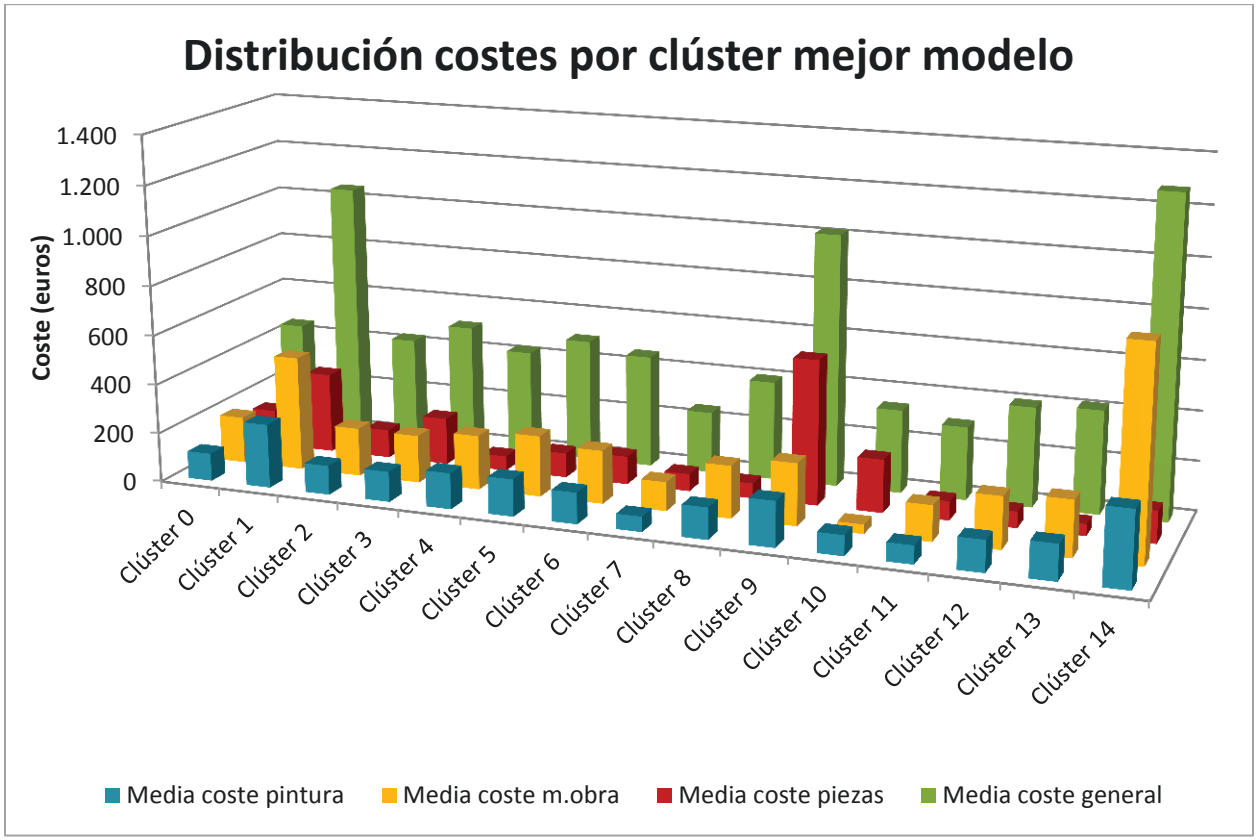


Gráfica 21: Auditoría de daños - EM ampliado. Comparativa atributos positivos mejor modelo

De nuevo se hace uso de estas gráficas comparativas (Gráfica 20 y Gráfica 21) para remarcar la distribución de instancias y atributos con porcentaje de positivos mayores del 10%. En esta ocasión, pese a presentar una distribución de instancias parecida a la mejor de la sección anterior (en otro orden), los porcentajes de positivos están mucho más balanceados (en comparación con la Gráfica 19), lo que significa que los clústeres no presentan tanta variedad de atributos, están más caracterizados.

Otro de los parámetros que puede caracterizar un clúster es sin duda los costes asociados. Las instancias contenidas en cada grupo presentan una tendencia de coste que también aporta información sobre el conjunto. Por ejemplo, en el caso de que se repitan tipos de impacto en clústeres diferentes, los costes pueden determinar la intensidad del impacto, con lo que se puede establecer una distinción. Además, teóricamente los clústeres que acumulen mayores positivos en atributos correspondientes a pintura, tendrán mayor coste de pintura, aquellos con mayores reparaciones acumularán coste de mano de obra y los que aglutinen mayoritariamente atributos de sustitución tendrán un alto coste en piezas.

Echando un vistazo a los costes asociados al mejor modelo (Gráfica 22), se observa que los clústeres que presentan una media de coste más alta son, curiosamente, lo mismos que presentan un mayor número de atributos con porcentaje de positivos mayor del 10%, como se aprecia en la gráfica anterior. Esta circunstancia parece lógica dado que las reparaciones más costosas son aquellas que implican un mayor número de piezas, o que incluyen las piezas más caras. Sea como fuere, sin duda los clústeres 1, 9 y 14 deben de corresponderse a los impactos de mayor intensidad, que son los más costosos de reparar.



Gráfica 22: Auditoría de daños - EM ampliado. Distribución costes por clúster mejor modelo

Otro hecho interesante es la diferencia de costes desglosados en los clústeres con mayor coste general. Mientras que los clústeres 1 y 14 presentan el coste mayoritario en mano de obra, el número 9 destaca por coste de piezas, por lo que es probable que dicho grupo implique un golpe sobre piezas internas. Según el desglose de atributos, las piezas internas correspondientes al motor no son ni reparables ni pintables.

Una vez realizadas las pesquisas necesarias sobre el conjunto, se procede a realizar un recuento detallado de las piezas contenidas en cada clúster. Por desgracia, las tablas resultantes no pueden visualizarse en este documento debido a su descomunal tamaño, por lo que se ha realizado una tabla resumen (Tabla 38), que aglutina las características más importantes del modelo. Se han interpretado los diferentes tipos de impacto atendiendo a las piezas más comunes y las partes clave que se detallan a continuación:

Clúster	% positivos > 10%	Coste general (€)	Atributos característicos	Partes clave	Tipo de impacto
0	9	447,26	PT_50101, PT_63203, REP_50101, REP_63203, SUST_50101, SUST_94101	Aleta delantera izquierda, Paragolpes delantero, faro delantero izquierdo	**Impacto delantero izquierdo**
1	29	1.050,35	PT_63203, PT_63303, PT_53101, PT_50101, PT_53102, PT_50102	Paragolpes delantero y trasero, Aletas delanteras izquierda y derecha, Aletas traseras izquierda y derecha	**Impacto frontal y trasero (colisión múltiple)**
2	12	429,63	PT_63303, PT_55303, SUST_66303, REP_63303, REP_55303, SUST_63303	Paragolpes trasero, Portón trasero, Anagrama del fabricante	**Impacto trasero central**
3	9	506,90	PT_50102, PT_63203, REP_50102, REP_63203, SUST_50102, SUST_94102	Aleta delantera derecha, Paragolpes delantero, Faro delantero derecho	**Impacto delantero derecho**
4	7	425,24	PT_58101, SUST_66201, REP_58101, PT_53101, REP_53101, PT_63303	Puerta trasera izquierda, Molduras izquierda, Aleta trasera izquierda, Paragolpes trasero	**Impacto trasero izquierdo (5 puertas)**
5	14	496,20	PT_57102, REP_57102, SUST_66202, PT_50102, PT_53102, REP_50102	Puerta delantera derecha, Molduras derecha, Aletas delantera y trasera derecha	**Impacto lateral derecho**
6	12	455,25	PT_57101, REP_57101, SUST_66201, PT_50101, REP_50101, PT_53101	Puerta delantera izquierda, Aletas delantera y trasera izquierda, Molduras izquierda, Retrovisor izquierdo	**Impacto lateral izquierdo**
7	4	248,91	PT_63303, REP_63303, SUST_63303, SUST_66953	Paragolpes trasero, Matrícula	**Impacto trasero central leve**
8	7	400,25	PT_53102, REP_53102, PT_63303, REP_63303, SUST_66202, SUST_63303	Aleta trasera derecha, Paragolpes trasero, Molduras derecha, Puerta trasera derecha	**Impacto trasero derecho (3 puertas)**
9	18	1.017,19	PT_63203, SUST_63203, PT_55103, SUST_50203, SUST_66103, SUST_94101	Paragolpes delantero, Capó, Coraza frontal, Rejilla del radiador, Faros delanteros	**Impacto delantero central severo**

10	2	337,41	SUST_64103, PT_51503	Parabrisas, Techo	**Impacto en parabrisas-techo (caída de objeto)**
11	6	295,92	PT_63203, REP_63203, SUST_66953, PT_55103, SUST_63203, REP_55103	Capó, Paragolpes delantero, Matrícula	**Impacto delantero central leve**
12	8	401,94	PT_53101, PT_63303, REP_53101, REP_63303, SUST_66201, SUST_94201	Aleta trasera izquierda, Paragolpes trasero, Molduras izquierda, Faro trasero izquierdo	**Impacto trasero izquierdo (3 puertas)**
13	7	416,43	PT_58102, REP_66202, REP_58102, PT_53102, REP_53102, REP_63303	Puerta trasera derecha, Molduras derecha, Aleta trasera derecha, Paragolpes trasero	**Impacto trasero derecho (5 puertas)**
14	32	1.275,61	PT_53102, PT_53101, PT_57102, PT_57101, PT_55103, PT_63303	Paragolpes delantero y trasero, Aletas delanteras izquierda y derecha, Aletas traseras izquierda y derecha, Puertas delanteras izquierda y derecha, Techo, Múltiples partes	**Impacto por todo el vehículo (vueltas de campana)**

Tabla 38: Auditoría de daños - EM ampliado. Desglose detallado mejor modelo

La tabla pone de manifiesto que el proceso de *clustering* ha cumplido bien con su cometido, las zonas y tipos encontrados aparentemente están delimitados, y parece que abarcan todas las posibles áreas de efecto de una colisión.

Tras el análisis exhaustivo, se han observado ciertas circunstancias que conviene destacar:

- El agrupamiento ha sido lo suficientemente preciso como para establecer diferentes tipos de colisión trasera lateral atendiendo a si el vehículo tiene 3 o 5 puertas. Se desconocía *a priori* si existían instancias de vehículos con 3 puertas.
- El algoritmo también ha establecido distinción entre distintas intensidades de impacto delantero y trasero central. Como puede observarse, los impactos severos conllevan la reparación de más piezas, siendo el caso del choque delantero especialmente destacable porque conlleva la reparación de piezas del motor que hace que se dispare el coste (clúster 9).
- Es curioso que haya tantas ocurrencias de siniestros del tipo impacto en parabrisas-techo, que copan la distribución con el 19% del total de instancias.
- Los siniestros con mayor coste general son, además, los choques que pueden ser catalogados como más graves, como la colisión múltiple (clúster 1), el choque frontal severo (clúster 9), y la vuelta de campana (clúster 14).
- El *clustering* ha sido lo suficientemente completo como para abarcar todas las zonas posibles del vehículo (frontal, trasera, laterales, esquinas y techo).

Se considera que el resultado del análisis es satisfactorio, y que el modelo obtenido refleja de una manera bastante fidedigna todas las posibles tipologías de impacto que pueden darse en una colisión. Se trata de un modelo relativamente ligero, suficientemente preciso y sencillo.

En la siguiente página se puede observar de manera gráfica la distribución de zonas y tipologías de impacto (Ilustración 22).

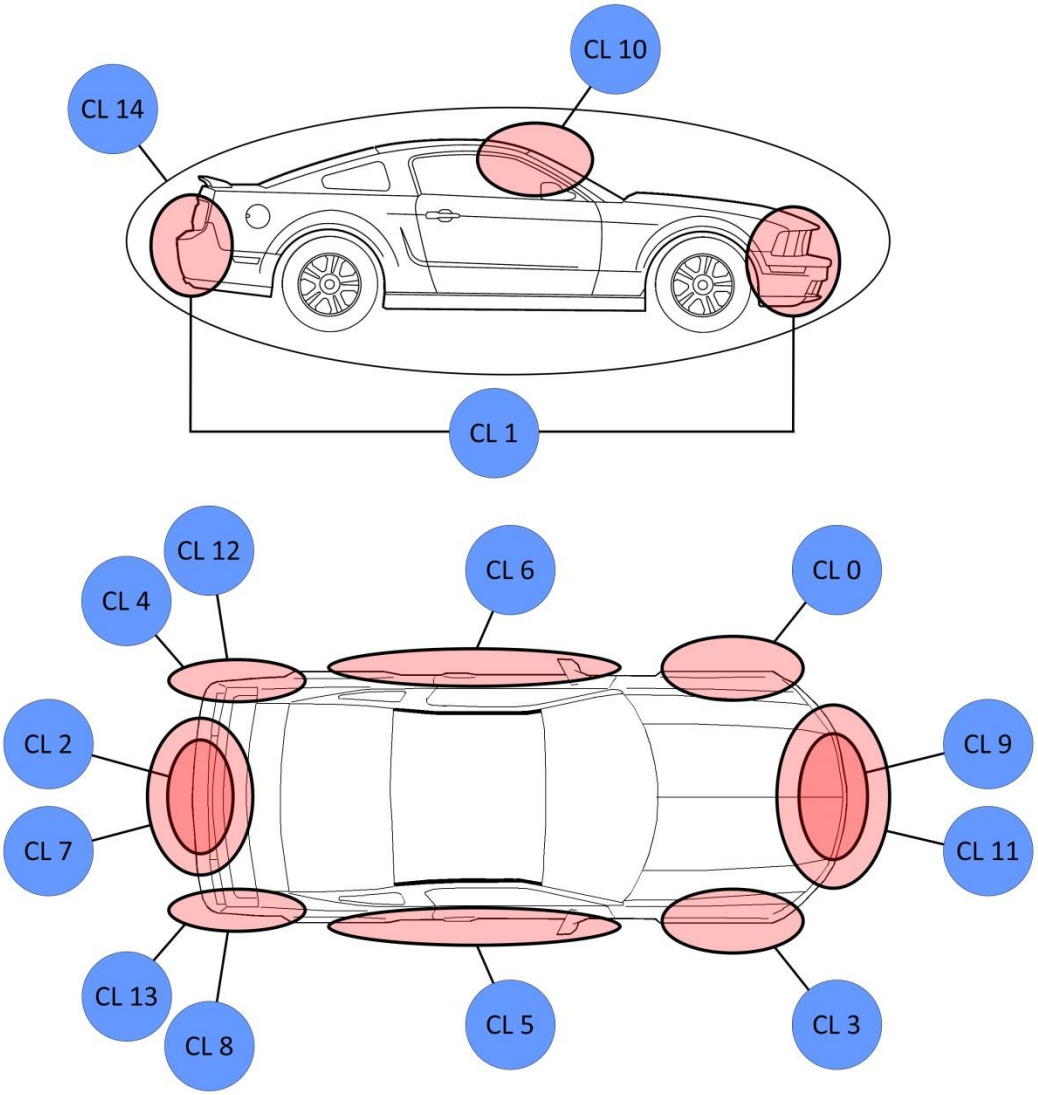


Ilustración 22: Auditoría de daños – Diagrama de tipos de impacto

Asignación de clúster a tipo de impacto			
Clúster 0	Impacto delantero izquierdo	**Clúster 8**	Impacto trasero derecho (3 puertas)
Clúster 1	Impacto frontal y trasero (colisión múltiple)	**Clúster 9**	Impacto delantero central severo
Clúster 2	Impacto trasero central	**Clúster 10**	Impacto en parabrisas-techo (caída de objeto)
Clúster 3	Impacto delantero derecho	**Clúster 11**	Impacto delantero central leve
Clúster 4	Impacto trasero izquierdo (5 puertas)	**Clúster 12**	Impacto trasero izquierdo (3 puertas)
Clúster 5	Impacto lateral derecho	**Clúster 13**	Impacto trasero derecho (5 puertas)
Clúster 6	Impacto lateral izquierdo	**Clúster 14**	Impacto por todo el vehículo (vueltas de campana)
Clúster 7	Impacto trasero central leve		

Tabla 39: Auditoría de daños – Asignación de clúster a tipo de impacto mejor modelo

5. Conclusiones y trabajos futuros

Tras la extensa experimentación llevada a cabo, los resultados globales de este proyecto se consideran muy satisfactorios, habiendo alcanzado los objetivos iniciales establecidos y pudiendo elaborar una amplia documentación al respecto. Ambas alternativas planteadas en el estudio consiguen aportar, desde dos enfoques diferentes, algunas soluciones al conocimiento global de la lucha contra el fraude en el seguro del automóvil.

5.1.Conclusiones del estudio

Al comenzar el proyecto se tomó la decisión de realizar una serie de pruebas iniciales para extraer toda la información posible sobre la formación del conjunto de datos y el agrupamiento inicial. Dado que el procesamiento previo al proyecto consistía en un *clustering* EM, se decidió emular el procedimiento. El test fue satisfactorio, se obtuvieron unos resultados similares a los que ya se tenían y se amplió el conocimiento sobre el conjunto de datos.

En este momento se decidió realizar una separación de tareas, dado que se vislumbraron dos posibles enfoques diferentes al problema: por un lado se continuaría con un análisis más en profundidad sobre la alternativa del estudio de la severidad de los partes de accidente, y por otro lado se comenzaría una investigación radicalmente distinta sobre las tipologías de impacto (auditoría de daños).

En cuanto a la alternativa de análisis de severidad, la experimentación sobre el *dataset* inicialmente resultó algo incoherente, los algoritmos de aprendizaje no supervisado utilizados no devolvían resultados similares. Un análisis más en profundidad y la inclusión de las pruebas 1R revelaron las claves, pudiendo establecer una relación entre los clústeres obtenidos y los grados de severidad que posteriormente centrarían el análisis. Las categorías derivadas del agrupamiento inicial fueron 4: grado 1 o severidad leve, grado 2 o severidad moderada, grado 3 o severidad alta y grado 4 o severidad muy alta. En añadidura, también se consiguió conocer la ganancia de información de los atributos utilizados para el agrupamiento.

Posteriormente, se procedió a realizar pruebas de aprendizaje supervisado sobre el *dataset* que permitieran obtener un conjunto de reglas de decisión. Estas reglas servirían para contrastar tasaciones y verificar si son clasificadas en el grado de severidad correcto. Se realizaron diferentes experimentos con el algoritmo de clasificación J48, obteniendo como mejor resultado un 96% de acierto de clasificación sobre un *dataset* con atributos de coste exclusivamente. Otras pruebas sobre *datasets* basados en los atributos binarios obtuvieron también excelente resultados, con un 77% de acierto de clasificación. De esta fase se obtuvieron dos modelos de conocimiento funcionales, uno basado en costes desglosados y el otro basado en piezas reparadas.

A tenor de toda la experimentación realizada, se puede afirmar que los análisis de severidad obtienen un mejor rendimiento si se basan en el coste de las reparaciones, y que éste es el principal agente clasificador, pero sin embargo, es posible llegar a la misma clasificación por otras vías, atendiendo exclusivamente al número de piezas reparadas y sabiendo cuales han sido éstas. Por ello, se han decidido mantener los dos modelos, el basado en atributos de coste y el basado en atributos binarios. El primer modelo presenta mayor tasa de acierto y es más sencillo, sin embargo, el segundo es

más versátil y robusto, haciendo más difícil que pueda ser sorteado por un ente con intención fraudulenta.

En cuanto a la auditoría de daños, el enfoque de esta alternativa implicó una aproximación distinta a la adoptada para el análisis de severidad. El establecimiento de un mapa de zonas de impacto y tipologías de colisión requirió de un nuevo proceso de *clustering* sobre el conjunto de datos, ya que el agrupamiento previo sólo se basaba en los atributos de coste. Para este procedimiento se utilizó de nuevo el algoritmo EM, más eficiente para este tipo de problema que otros algoritmos de aprendizaje no supervisado.

En las primeras pruebas lo resultados no fueron del todo satisfactorios, dejando sin embargo entrever prometedores avances. Posteriormente, la inclusión de un *dataset* de mayor tamaño y el afinamiento de los parámetros de configuración del algoritmo empezaron a dar sus frutos, logrando establecer un número suficiente de clústeres como para cubrir todas las tipologías de impacto representadas en los ejemplos del conjunto de datos. La inclusión del *dataset* de 151.826 instancias permitió la obtención de un modelo fidedigno de tipologías de impacto, con 15 clústeres que representan 15 tipos diferentes de colisión y una función de verosimilitud de -9.41042. Los clústeres demostraron contener conjuntos de atributos binarios disjuntos entre sí en gran medida, lo que permitía “caracterizar” cada clúster como un tipo de colisión diferente.

Los experimentos realizados permiten concluir que el *dataset* contiene ejemplos que pueden ser clasificados en 15 tipos de impacto diferentes, que abarcan todas las zonas del coche susceptibles de ser dañadas en un siniestro y además distinguen entre diferentes cualidades del vehículo en cuestión, como el número de puertas (3 o 5 puertas) y la intensidad de ciertos tipos de colisiones, como la frontal o la trasera. Se considera que el modelo obtenido es muy bueno, fruto de una experimentación meticulosa y concienzuda.

En lo personal, el proyecto ha resultado de lo más atractivo, con un nivel de exigencia importante debido a los objetivos que se intentaban acometer. En primer lugar, la elaboración de la sección de Estado del Arte y la investigación necesaria para afrontar la experimentación han resultado muy enriquecedoras, dado que la temática del proyecto abarca un espectro bastante grande. Sin duda, conocer toda la maquinaria que rodea la peritación, la lucha contra el fraude y el sector de automoción es muy interesante.

En cuanto al desarrollo de la experimentación, pese a que la utilización de algoritmos de *data mining* sobre conjuntos de datos tan grandes ha supuesto largas horas de procesamiento y multitud de repeticiones, los resultados obtenidos se consideran de lo más satisfactorios. Este proyecto no podría haberse acometido sin una experimentación minuciosa y precisa.

5.2.Trabajos futuros

En adición a las conclusiones, se pueden establecer algunas ideas para líneas futuras de trabajo relacionadas con este proyecto:

Como ya se describió en el planteamiento del proyecto, este estudio no permite la clasificación de instancias de siniestro como fraudulentas o no, únicamente ejerce de ente evaluador que pretende aportar información y conocimiento, y que posteriormente un tercer agente sea el que realice la catalogación de futuras instancias como fraudulentas o no. Esa puede ser precisamente una de la líneas futuras de este trabajo, ampliar el alcance del mismo creando un mecanismo clasificador de instancias fraudulentas utilizando los modelos de conocimiento extraídos en este proyecto. Para este trabajo, sin embargo, será necesario en primer lugar disponer de un conjunto de datos con instancias fraudulentas contrastadas, a fin de poder supervisar las decisiones del clasificador y evaluar su rendimiento.

En la auditoría de daños, queda quizá pendiente la elaboración de un modelo basado en reglas de decisión que permita clasificar tasaciones automáticamente asignándolas una tipología de impacto concreta. Esta línea de trabajo sería bastante interesante, y podría ser implementada en conjunción con otras nuevas tecnologías nuevas del mercado, como el *ecall*. Ésta nueva tecnología consiste en instalar un dispositivo en el vehículo que realiza una llamada a los servicios de emergencia al producirse un incidente, y envía a su vez una serie de datos acerca del siniestro y del vehículo implicado. Si estos datos se amplían y se pudiera elaborar un informe de daños completo, podría a su vez introducirse en un clasificador basado en el modelo obtenido en este proyecto y completar un informe de peritación de manera automática, con la consecuente ventaja de la rapidez y eficiencia para la compañía de seguros. La función del perito, en este caso, se vería reducida a una mera supervisión del proceso, y eliminaría sin duda riesgos de fraude al contar el proceso con un factor de control mucho mayor.

El análisis de severidad, por otra parte, presenta una alternativa bastante interesante para las compañías aseguradoras, y es la inclusión en la relación entre la propia compañía y el taller concertado de lo que podría denominarse como Tarifas Planas de Tasación. Estas tarifas planas consistirían en establecer costes medios para los grados de severidad de un siniestro, y aplicar este coste a todas las tasaciones que sean catalogadas en cada una de las categorías de severidad. De esta manera, se eliminan posibles fraudes realizados en la peritación o el presupuestado, garantizando además que las tasaciones son justas, dado que si en una ocasión el precio de la reparación supera la media, con la consiguiente pérdida de dinero por parte del taller, en la siguiente tasación puede ocurrir lo contrario, suponiendo un balance entre ganancias y pérdidas. En añadidura, el proceso de valoración de siniestros se aceleraría sobremanera, dado que se eliminarían algunas etapas y procesos burocráticos.

6. Planificación

En esta sección se expone la planificación detallada del desarrollo total del proyecto, desde la planificación inicial, elaborada al comenzar el mismo para establecer las pautas e hitos a seguir, a la planificación final corregida y ajustada al desarrollo real del proyecto.

La primera planificación se concibió de una manera conservadora, planeando con holgura el tiempo y esbozando únicamente las fases principales del proyecto, dado que no se conocía el alcance total del trabajo. Se planificaron 5 fases principales, más una fase de contratiempos y el día de la propia presentación. El diagrama Gantt correspondiente se puede consultar en el Anexo E (Ilustración 25: Planificación inicial). Las fases son las siguientes:

- Fase inicial: Que abarca desde el principio del proyecto hasta el establecimiento del entorno y la familiarización con los elementos del proyecto. Duración de 8 días.
- Investigación: Etapa de formación y búsqueda sobre documentación relacionada con el ámbito del proyecto. Duración de 30 días.
- Experimentación: Desarrollo de las pruebas relativas a la solución planteada para el problema. Duración de 30 días.
- Elaboración de la memoria: Elaboración de este documento, que refleja todos los aspectos abordados en la realización del proyecto. Duración de 30 días.
- Revisión general: Un repaso de todo el trabajo con el fin de pulir detalles y corregir pequeños errores. Duración de 7 días.

La planificación inicial preveía una duración total del proyecto de 103 días más 15 días de contratiempos, lo que hacen un total de 118 días. La previsión de trabajo se limitó a 4 horas diarias, lo que hacen un total de **412 horas de trabajo**, sin incluir los días de contratiempos.

Finalmente, el proyecto necesitó de todo el tiempo extra para contratiempos debido a que se dividió la experimentación en las 2 alternativas de análisis, alargándose cada una de ellas 30 y 27 días respectivamente. Este hecho también requirió que en ciertas partes se alternaran dos fases de manera solapada, como en el caso de la experimentación y la elaboración de la memoria. Debido a esta circunstancia, en la planificación final ha sido eliminada la fase de contratiempos, que ha sido absorbida por las otras etapas del desarrollo.

Las partes de mayor duración han sido la experimentación, con 69 días, y la elaboración de la memoria, que ha durado 98 días debido al solapamiento con la experimentación y la necesidad de acabar ciertas tareas antes de la continuación con otras partes de la memoria. Por último, algunas partes se han contraído, como es el caso de la revisión, que únicamente ha durado 3 días.

La planificación final refleja una duración total del proyecto de 116 días, con una media de 3 horas trabajadas en las primeras fases del desarrollo (66 días) y 5 horas trabajadas por día en las fases finales (50 días), lo que hacen un total de **448 horas de trabajo totales**. La planificación final completa se puede consultar en el Anexo E (Ilustración 26: Planificación final).

7. Presupuesto

A continuación se presenta la estimación detallada de costes de desarrollo en términos de personal, equipamiento y licencias de software.

Para el cálculo de costes imputables para equipamiento y licencias, se ha utilizado un periodo redondeado de duración del proyecto de 4 meses, además del tiempo de amortización desde la fecha de adquisición del equipo/software.

- Autor: Eduardo González González
- Departamento: Departamento de informática
- Proyecto: Auditoría de datos aplicado a siniestros de automóviles
- Desglose:

PERSONAL				
Nombre	Categoría	Sueldo por hora	Horas trabajadas	Coste total
Eduardo González González	Investigador principal	20,00 €/h	448 h	8960,00 €
			Total (€):	**8960,00 €**

EQUIPAMIENTO					
Descripción	Coste	Duración del proyecto	Uso dedicado al proyecto	Periodo de amortización	Coste imputable
PC Sobremesa Medion 8340	400 €	4 meses	50%	36 meses	22,20 €
PC Portátil Lenovo Z500	780 €	4 meses	70%	22 meses	99,30 €
				Total (€):	**121,50 €**

SOFTWARE					
Descripción	Coste	Duración del proyecto	Uso dedicado al proyecto	Periodo de amortización	Coste imputable
Licencia Windows 8.1 (portátil)	119 €	4 meses	70%	22 meses	15,10 €
Microsoft Office 2010 Professional	307 €	4 meses	60%	36 meses	20,50 €
Adobe Photoshop CS6	927 €	4 meses	20%	15 meses	49, 40 €
Notepad ++	0 €	4 meses	50%	-	0 €
WEKA	0 €	4 meses	100%	-	0 €
Gantter	0 €	4 meses	100%	-	0 €
				Total (€):	**85,00 €**

Tabla 40: Presupuesto - Desglose de costes

RESUMEN	
Concepto	Coste
Personal	8960,00€
Equipamiento	121,50€
Software	85,00€
Total sin IVA	9166,50€
IVA (21%)	1924,96€
Total	**11091,46€**

Tabla 41: Presupuesto - Coste total del proyecto

8. Bibliografía

Bajo formato IEEE:

[1] Real Academia Española de la Lengua (RAE), «perito, ta,» [En línea]. Available: http://lema.rae.es/drae/srv/search?id=Pam1BpoVz2x5szAn6Tc. [Último acceso: 15 Abril 2015].

[2] Arpem Networks S.L., «definición y responsabilidades del perito,» 2015. [En línea]. Available: http://www.arpem.com/seguros/glosario/perito.html. [Último acceso: 10 Mayo 2015].

[3] Arpem Networks S.L., «Ley 50/1980, de 8 de octubre, de Contrato de Seguro. Sección I: Disposiciones Generales,» 1980. [En línea]. Available: http://www.arpem.com/varios/legislacion/seguros/ley5080/l50-1980.t2.html#a38. [Último acceso: 23 Mayo 2015].

[4] Audatex, «Servicios de valor añadido Audatex,» [En línea]. Available: http://www.audatex.es/audatexServices. [Último acceso: 22 mayo 2015].

[5] GT Estimate, «Sobre GT Motive,» [En línea]. Available: http://gtmotive.com/es/empresa/sobre-gt-motive. [Último acceso: 23 Abril 2015].

[6] Asociación Nacional de Vendedores de Vehículos a Motor, Reparación y Recambios (GANVAM), «Website de GANVAM: ¿Qué es GANVAM?,» [En línea]. Available: http://www.ganvam.es/ganvam/que-es. [Último acceso: 26 Abril 2015].

[7] Tecnologías de la Información y Redes para las Entidades Aseguradoras (TIREA), «Descripción del Fichero Histórico del Seguro del Automóvil (SINCO),» [En línea]. Available: http://www.tirea.es/Entidades-Aseguradoras/Autos/Sinco/Descripcion.aspx. [Último acceso: 10 Mayo 2015].

[8] EurotaxGlass's, «Eurotax: Sobre nosotros,» [En línea]. Available: http://www.eurotax.es/sobre-nosotros/. [Último acceso: 25 Mayo 2015].

[9] Centro de Estudios del Consejo General de los Colegios de Mediadores de Seguros, «Website del CECAS,» [En línea]. Available: http://www.cibercecas.com/. [Último acceso: 2015 Mayo 20].

[10] M. Ayuso Gutiérrez y M. Guillén Estany, Modelos econométricos para la detección del fraude en el seguro del automóvil, Tesis doctoral. Barcelona: Universidad de Barcelona, 1998.

[11] M. Iturgoyen y MAPFRE Mutualidad, «La tipología del fraude en el seguro de automóviles de España,» 1996. [En línea]. Available: http://www.mapfre.com/documentacion/publico/i18n/catalogo_imagenes/grupo.cmd?path=1035854. [Último acceso: 24 Mayo 2015].

[12] Línea Directa, «2º Barómetro Línea Directa 2013: El Fraude en el seguro de autos,» Madrid, 2013.

[13] Axa Seguros, «II Mapa AXA del fraude en España Marzo 2015,» 2015.

[14] Investigación Cooperativa entre Entidades Aseguradoras y Fondos de Pensiones (ICEA), «El Fraude al Seguro Español. Estadística año 2014,» Madrid, 2014.

[15] M. Ayuso Gutiérrez y M. Santolino, «Una revisión metodológica de la valoración actuarial de los siniestros con daños corporales en el seguro del automóvil,» *Anales del Instituto de Actuarios Españoles,* vol. 3, nº 13, 2007.

[16] Investigación Cooperativa entre Entidades Aseguradoras y Fondos de Pensiones (ICEA), «Tipologías de fraude en seguros,» 23 Octubre 2013. [En línea]. Available: http://www.icea.es/es-es/noticias/Noticias/Noticias1013/Dia_23/fraude-seguros.aspx?UrlVolver=%2Fes-es%2Fnoticias%2Fpaginas%2Fultimasnoticias.aspx%3Ftema%3Dlucha%2520contra%2520el%2520fraude. [Último acceso: 24 Mayo 2015].

[17] R. Kohavi y F. Provost, «Glossary of Terms - Special Issue on Applications of Machine Learning and the Knowledge Discovery Process,» *Machine Learning,* vol. 30, nº 2-3, pp. 271-274, 1998.

[18] I. H. Witten y E. Frank, Practical Machine Learning Tools and Techniques, San Francisco (USA): Morgan Kaufmann, 2005.

[19] F. Gurunescu, «Introduction to Data Mining,» de *Data Mining - Concepts, models and techniques*, Craiova (Rumanía), Springer - Verlag Berlin Heidelberg, 2011, pp. 1-46.

[20] D. Pregibon, «Data Mining,» *Statistical Computing & Graphics Newsletter,* vol. 7, nº 3, p. 8, 1996.

[21] J. W. Tukey, «The Future of Data Analysis,» *The Annals of Mathematical Statistics,* vol. 33, nº 1, pp. 1-67, 1962.

[22] Real Academia Española de la Lengua (RAE), «base de datos,» [En línea]. Available: http://lema.rae.es/drae/?val=base+de+datos. [Último acceso: 23 Abril 2015].

[23] Encyclopaedia Britannica Online, «machine learning,» Encyclopaedia Britannica, [En línea]. Available: http://global.britannica.com/EBchecked/topic/1116194/machine-learning. [Último acceso: 22 Mayo 2015].

[24] L. C. Molina Félix, «Data mining: torturando a los datos hasta que confiesen,» *UOC,* vol. I, nº 1, pp. 1-3, 2002.

[25] Usuarios de Wikipedia, «machine learning,» Wikipedia The Free Encyclopedia, 2015. [En línea]. Available: http://en.wikipedia.org/wiki/Machine_learning. [Último acceso: 10 Abril 2015].

[26] G. Piatetsky y A. Rajpurohit, «The Cardinal Sin of Data Mining and Data Science: Overfitting,» *KD Nuggets: Data Mining, Analytics, Big Data, and Data Science,* vol. 14, nº 15, 18 Junio 2014.

[27] J. R. Quinlan, C4.5: Programs for Machine Learning, San Francisco (USA): Morgan Kaufmann, 1993.

[28] C. Bielza y P. Larrañaga, *Transparencias de la asignatura Aprendizaje Automático: Árboles de Clasificación,* Madrid: Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid, 2012.

[29] R. C. Holte, «Very Simple Classification Rules Perform Well on Most Commonly Used Datasets,» *Machine*

Learning, vol. 11, nº 1, pp. 63-91, 1993.

[30] J. B. MacQueen, «Some Methods for classification and Analysis of Multivariate Observations,» de *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. 1. University of California Press*, Berkeley (USA), 1967.

[31] A. A. Adebisi, O. E. Olusayo y O. S. Olatunde, «An Exploratory Study of K-Means and Expectation Maximization Algorithms,» *British Journal of Mathematics & Computer Science,* vol. 2, nº 2, pp. 62-71, 2012.

[32] A. P. Dempster, N. M. Laird y D. B. Rubin, «Maximum Likelihood from Incomplete Data via the EM algorithm,» *Journal of the Royal Statistical Society, Series B,* vol. 39, nº 1, pp. 1-39, 1977.

[33] M. Pejic-Bach, «Profiling intelligent systems applications in fraud detection and prevention: survey of research articles,» de *2010 International Conference on Intelligent Systems, Modelling and Simulation*, Liverpool (UK), 2010.

[34] R. A. Derrig y K. M. Ostaszewski, «Fuzzy Techniques of Pattern Recognition in Risk and Claim Classification,» *The Journal of Risk and Insurance,* vol. 62, nº 3, pp. 447-482, 1995.

[35] S. Viaene, R. A. Derrig y G. Dedene, «A Case Study of Applying Boosting Naive Bayes to Claim Fraud Diagnosis,» *IEEE Transactions on Knowledge & Data Engineering,* vol. 16, nº 5, pp. 612-620, 2004.

[36] G. Facchinetti y S. Bordoni, «Insurance Fraud Evaluation: A fuzzy expert system,» *The 10th IEEE International Conference on Fuzzy Systems,* vol. 3, pp. 1491,1494, 2001.

[37] Yi Peng, Gang Kou, A. Sabatka, D. Khazanchi y Yong Shi, «Application of Clustering Methods to Health Insurance Fraud Detection,» *2006 International Conference on Service Systems and Service Management,* vol. 1, pp. 116-120, 2006.

[38] S. Hajian, J. Domingo-Ferrer y A. Martínez-Ballesté, «Discrimination prevention in data mining for intrusion and crime detection,» *2011 IEEE Symposium on Computational Intelligence in Cyber Security (CICS),* pp. 47-54, 2011.

[39] S. P. D'Arcy, «Predictive Modeling in Automobile Insurance: A Preliminary Analysis,» de *World Risk and Insurance Economics Congress*, Salt Lake City (USA), 2005.

[40] J. H. Yoo, B. H. Kang y J. U. Choi, «A hybrid approach to autoinsurance claim processing system,» *Systems, Man and Cybernetics,* vol. 1, pp. 537-542, 1994.

[41] W. K. Tseng y C. S. Lu, «The system for appraisal of vehicle accident based on radial basis function neural networks,» *2011 Seventh International Conference on Natural Computation (ICNC),* vol. 2, pp. 869-872, 2011.

[42] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann y I. H. Witten, «The WEKA Data Mining Software: An Update,» *SIGKDD Explorations,* vol. 11, nº 1, 2009.

[43] Q. Isaac y A. Simón, *Transparencias de Diseño de Experimentos para el Reconocimiento de Patrones,* Valladolid: Curso de doctorado, Universidad de Valladolid, 2004.

ANEXO A: English summary

Abstract

Given the increasing importance that fight against fraud in car insurance is currently taking, it is necessary to develop new assessment tools that complement traditional methods of insurance claim analysis.

This project aims to conduct a detailed modeling of the essential characteristics of a car accident, such as type, impact direction or severity degree, in order to automate the appraisal process and to facilitate detection of fraudulent or erroneous claims. For this procedure, expert reports of past accidents provided by an important spanish claim management company will be taken into account.

During the development of this paper, various data mining and machine learning techniques, both supervised and unsupervised, will be used. These methods will be applied over a data file containing multiple instances of damage appraisals. Different knowledge models will be extracted in the testing process, in order to fulfill the general purpose of the project.

Keywords

Car accident, claim appraisal, fraud, data analysis, data mining, machine learning, supervised learning, unsupervised learning, clustering

Introduction

Nowadays, the Internet generates, every second, enormous amounts of data derived from every imaginable human activity, from the highway ticket containing a vehicle's plate to the hourly weather forecast in every urban area of the country, the photos uploaded to a personal profile from a social network or the latest trends in fashion from the celebrities. In this Digital Age, there is so much information hosted in servers across the globe that has become indispensable to design and plan a whole new branch of technologies to manage such an amount of data.

In recent years as well, awareness of the value of information has raised, being the industrial sector and the business world the biggest exponents of this growth. In this area information is considered a capital asset capable of being exploited, appreciated but also dangerous if not handled with care or neglected. Being well informed can be helpful when making decisions, optimizing, predicting or planning over a certain field, gaining an advantage which may result in an economic benefit.

The actual development of technology has provided powerful tools for handling data. Through different disciplines such as Statistics, Artificial Intelligence, Machine Learning or Data Mining, valuable knowledge can be extracted and used afterwards as a catalyst for better efficiency and performance.

On the other hand, the automobile industry has always been one of the economic engines of the developed countries due to the level of dependence on cars that human activities have reached. The Spanish automotive market is among the 5 most powerful within Europe, along with Germany, UK, France and Italy. Such turnover not only creates revenue through new vehicle sales, but also with second hand vehicle sales, spare parts, repairs, insurance and maintenance.

In the auto-insurance sector, having reliable and accurate information is considered vital when performing claim evaluations, appraisals and fraud analysis. An incorrect assessment affects both the company and the rest of insured clients, therefore is desirable to obtain optimum results in the analysis of the expert reports. Currently, the auto-insurance fraud constitutes more than 70% of all cases of swindle to insurance companies in Spain. Therefore, it is considered one of the most critical areas of the business, with a huge increase in investment in recent years.

The modern insurance policies, with never ending clauses and coverage combinations, added to the enormous size of actual customer databases, require new methods and tools to manage all that information. The old supervision techniques seem insufficient given the new demands of the sector, hence experts nowadays have access to a wide variety of tools to do their jobs, including advanced applications for insurance claim analysis, coding standards, scales or consulting services offered by third parties.

The human factor is undoubtedly the weakest link in the chain of insurance claim appraisal. The risks attributable to individuals not only depend on the honesty when making expert reports or assessing damage, but can also occur due to an excessive workload, with repetitive tasks that may cause human errors, or occurrence of new fraud techniques unknown to the experts. Technology can reduce these risks by progressively introducing more automated phases, increasing control.

This project aims to provide new tools and solutions for expert appraisal and damage assessment, considered the least "automatable" phases of the insurance claim processing system, due to their complexity and dependence on human knowledge.

Complementing the main purpose, the following specific objectives are proposed:

- Conduct a study of the essential characteristics of a car accident, taking into account expert reports and damage appraisals occurred in the past.
- Design a specific knowledge pattern that allows classifying expert evaluations according to the severity degree of the accident to assess.
- Create a damage audit model relating the most typical damage repairs in an appraisal to the vehicle's impact zones.

This project aims to design not a mechanism capable of classifying fraudulent appraisals, but a mere analyzer of the essential features of a claim, in order to provide information for third parties to take the decision.

Experimentation

Project structure

The project that is going to be undertaken is essentially defined as a detailed data analysis over a set of instances processed beforehand. The experimentation, which will use various data mining techniques methods and techniques, is specified below.

The file containing the data was subjected to a clustering process prior to the project, which returned a number of assignments stored in the studied dataset itself. These class attributes will act as "desired output" for supervised learning algorithms, in case these are used.

This paper is divided into two main approaches: severity analysis and damage audit. The severity analysis is based on the previous clustering results and aims to extend the tests and obtain knowledge models applicable in the fight against auto-insurance fraud, while damage audit focuses on performing a variety of tests over the given dataset in order to establish new data groups based on types of impact.

The first stage of the severity degree analysis consists of emulating the clustering process prior to the project with the purpose of gaining more information about the dataset and determining the categories that define the dataset distribution. Unsupervised learning algorithms will be used for this procedure, along with other algorithms that can provide a better understanding of the data. It will be necessary to preprocess the data before starting the testing phase, so as to suit the needs of the problem.

In a second phase, the study on the results of the clustering process is extended. Through supervised classification algorithms like C4.5, the purpose of this stage is to obtain a model based on decision rules that will be used to define the severity classification. With these decision rules, it is intended to classify future instances as accurately as possible. The model can be used to test insurance claims suspicious of being fraudulent.

As for the damage audit, it is planned to attain a whole new grouping by running a clustering process based in different parameters. This procedure will allow to discriminate collision types by considering the most typical repairs occurred in damage appraisals. To this end, various tests will be performed using clustering algorithms, such as Expectation-Maximization or K-means, aiming to achieve a completely different distribution. Ideally, those instances that present the most representative damages in each area of the vehicle will be grouped into similar clusters, which will ultimately shape a map of impact zones based on piece groups, location, intensity and trajectories.

Finally, conclusions about the two approaches will be drawn and a discussion over possible future applications of the project in the fight against fraud will be proposed.

Severity analysis

A typical risk of fraud lies in the damage assessment executed by providers to the insurance company. A claim may be evaluated incorrectly by the insurance expert looking to get some sort of benefit, implying a mismatch in the repair cost, which result on a loss for the company. Likewise, the budget presented by the repair shop may be inflated in order to increase benefits as well. In either case it is desirable to have mechanisms to contrast the appraisal and detect anomalies, if any.

One of the approaches proposed in this project is to design a knowledge model based on severity levels for classifying new claims automatically according to a set of decision rules.

With this system, an accident can be classified within one grade of severity (represented by a cluster). An appraisal can be double-checked by applying the obtained decision rules, verifying whether it is classified in the correct severity grade or not (Figure 1).

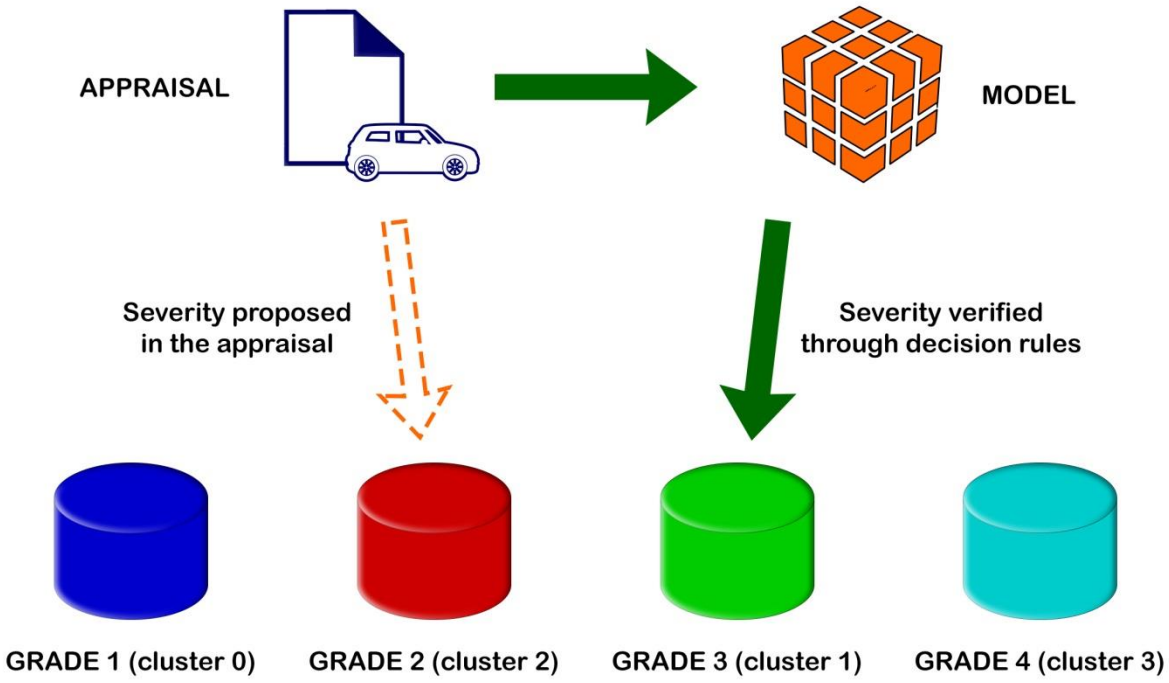


Figure 1: Severity analysis

As seen in the image, initially an appraisal may classify the claim with a specific grade of severity, depending on the expert's evaluation. In order to avoid erroneous or fraudulent assessments, a verification of the classification can be executed applying the knowledge model.

As a first option to obtain a set of decision rules, it is decided to execute different C4.5 test on the dataset, which will generate a decision tree for each test and a set of associated decision rules. The C4.5 tree is based on information gain ratio, in order to designate the different branch bifurcations. Therefore, attributes with higher information gain will be positioned at the highest nodes (root nodes), while those with less contribution will be located near the leaves. C4.5 allows to different pruning

methods which are Confidence Factor (default) and Reduced Error Pruning. Both pruning methods will be executed in every test, in order to obtain the best performance.

First, different tests will be executed over a dataset based only on cost attributes, which are:

- Tot_mo: Total direct labor cost of every instance.
- Tot_pint: Total painting cost of every instance.
- Tot_piez: Total piece replacement costs of every instance.

After several test, the results are the following:

Test	CF				REP			
	CF	minNumObj	Tree size	Accuracy	Folds	minNumObj	Tree size	Accuracy
1	0.25	2	313	98,87%	3	2	189	98,61%
2	0.05	50	65	97,66%	3	50	35	97,38%
3	0.01	100	47	97,20%	3	100	27	96,85%
4	0.005	150	25	96,86%	3	150	23	96,27%

Table 1: Severity analysis results. Cost attributes

The best effort is the one achieved by the last test, with enabled reduced error pruning. The obtained tree is simple and accurate enough for being considered as the winning model for this configuration.

A second set of tests will be performed over a different dataset, this time formed only by binary and piece counters, being the following:

- 105 binary attributes that represent painting, repairing and replacement operations.
- 2 counting attributes: Pos_int, which represents the number of replaced pieces, and Pos _mod, representing the number of repaired pieces.

The results for the tests on the second dataset are the following:

Test	CF				REP			
	CF	minNumObj	Tree size	Accuracy	Folds	minNumObj	Tree size	Accuracy
1	0.25	2	1783	84,55%	3	2	1453	84,40%
2	0.05	50	175	82,74%	3	50	151	81,85%
3	0.01	100	105	80,28%	3	100	97	78,68%
4	0.001	200	55	77,04%	3	150	61	77,13%

Table 2: Severity analysis results. Binary attributes

The chosen model is the one attained in the fourth test, with CF enabled this time. Despite showing only 77% accuracy, the size of the tree is very good considering the high dimensionality of the dataset. These two parameters are considered good enough for the analysis.

At the end, the experimentation reveled two valid models for this approach. Both will be taken into consideration in the conclusions.

Damage audit

As mentioned in different studies about auto-insurance fraud, 34% of fraudulent actions detected tally to the concealment of pre-existing damage in the vehicle, that is, while claiming over an accident, minor damage occurred in previous collisions is included in the report, so the insurance covers this damage and the fraudster avoids penalties established in the policy. This is, certainly, one of the most common fraud types perpetrated by the insured himself, and in most cases is difficult to detect.

An interesting alternative would be to develop a tool able to check the characteristics of a damage appraisal and detect which aspects do not fit within the overall scene of the accident. This evaluation will allow to dictate whether all damage is been caused in the accident or not.

This project aims to study and design a knowledge model that allows to group damage appraisals according to its impact typology (Figure 2), and to determine which parts are usually the most commonly repaired, replaced or repainted within the resulting types. If damage not corresponding to the type of collision is found, that claim would be labeled as fraudulent.

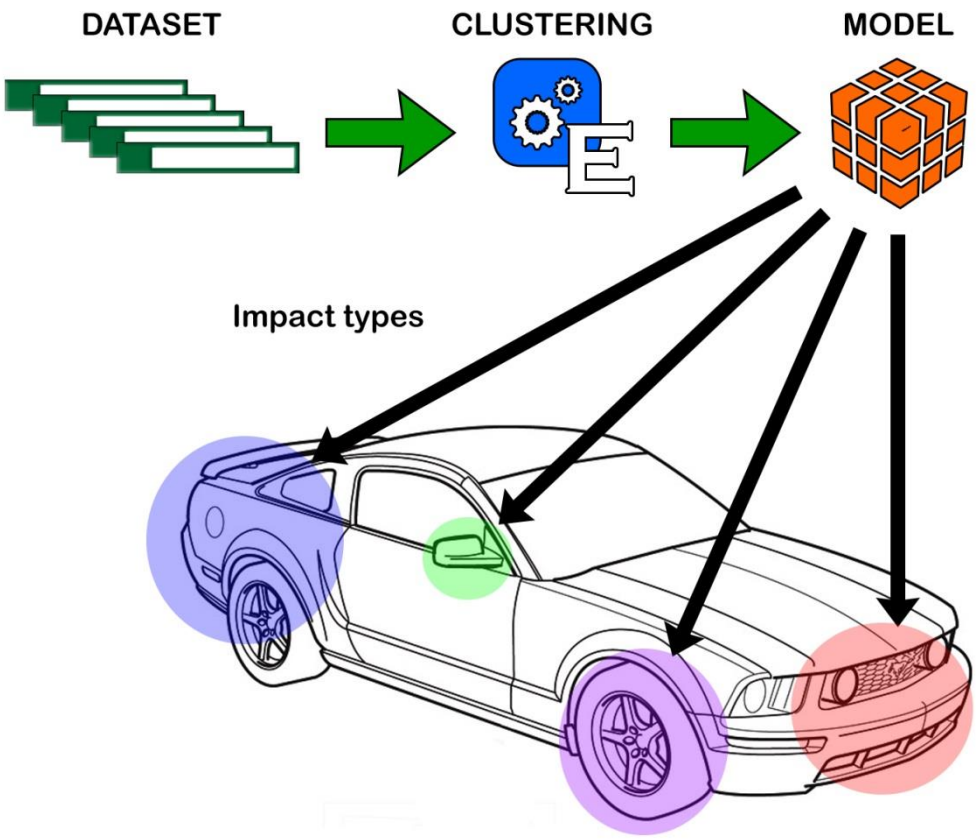


Figure 2: Damage audit

Instinctively, the first choice when trying to establish a distinction within a dataset is to execute a clustering process on the mentioned dataset. As indicated in the definition of clustering: “This procedure lies in grouping different objects, so that those who belong to a certain cluster resemble each other more than those present in other groups”.

The Expectation-Maximization algorithm has been chosen again for the clustering over K-means because ensures convergence in few iterations of the algorithm and also proves better performance when a fixed number of clusters is set for the computation instead of unknown number of clusters. Since types of impact are unknown from the start, the dataset will be progressively tested with different number of clusters in order to obtain the best results.

A new dataset has been composed for this experimentation, consisting of 151.826 instances instead of 27.706. This will allow a better convergence for the EM algorithm, considering the high dimensionality of the dataset (105 attributes).

The experimentation on this dataset will consist of several tests considering only binary attributes, which are:

- 105 binary attributes that represent painting, repairing and replacement operations.

After the procedure, the results are as follow:

Test	Number of clusters	Iterations	Log likelihood
1	10	50	-9.78503
2	12	62	-9.69458
3	15	38	-9.41042

Table 3: Damage audit results. Large dataset

The best effort was achieved in the last test, which involved a fixed number of 15 clusters. The model showed the highest log likelihood among all tests and the vehicle regions obtained covered all types of collisions, being accurate enough to distinguish even different intensities over a same impact area.

This model will be taken into consideration in the conclusion section.

Conclusions

After carrying out an extensive experimentation, the overall results of this project can be considered as very satisfactory, having achieved the goals set at the beginning and developed a complete documentation on the subject. Both approaches proposed in the study contribute to the fight against fraud in the auto-insurance sector, each making an effort from a different angle.

At the very beginning, the developing team decided to perform a series of initial tests to extract all possible information on the given dataset and the previous clustering. Since the project preprocessing consisted on a clustering based on the Expectation-Maximization algorithm, it was decided to emulate the procedure. The test was adequate, similar results to those already known were obtained and extensive knowledge about the dataset was acquired.

The initial tests allowed to conceive a separation of tasks, as two possible approaches to the problem were conceptualized. The first one would continue with a more detailed study on the clustering obtained and emphasize on the severity classification of damage appraisals. The second approach would begin a different research on impact typology (damage audit).

Regarding the severity analysis, dataset experimentation was initially somewhat inconsistent, as the supervised learning algorithms used were unable to return solid results. A more in-depth analysis and the inclusion of 1R tests revealed certain keys and allowed to establish a relationship between the clusters obtained and severity degrees, vital for further experimentation. Four categories were obtained from the research: grade 1 or mild severity, grade 2 or moderate severity, grade 3 or high severity and grade 4 or very high severity. In addition, information gain of every attribute used for the initial clustering was acquired.

Subsequently, supervised learning tests were run on the dataset in order to obtain a set of decision rules. These rules would allow to contrast appraisals and to check that these are classified in the correct severity degree. Different experiments with the J48 (C4.5) algorithm were performed, obtaining an excellent 96% classification accuracy as the best result over a dataset composed only by cost attributes. Other tests run over datasets consisting only in binary attributes also achieved great results, with a 77% classification accuracy. At the end of this phase, two completely functional knowledge models were obtained, one based on costs and the other based on repaired parts.

After all the experimentation, it can be said that severity analysis achieves a better efficiency if based on the cost of repairs, and that this is the main classifying agent. However, it is possible to reach the same classification by other attributes like the ones that represent repaired parts. Therefore, it was decided to keep the two models. The first model has a higher success rate and is simpler, yet the second one is more versatile and robust, making it harder for individuals with fraudulent intent.

As for the damage audit, this alternative involved a different approach to the one adopted for the severity analysis. Establishing a map of impact zones and types of collision required a new process of clustering over the dataset, as the previous processing was based only on cost attributes. For this

procedure the EM algorithm was used again, being more efficient for this type of problem than other unsupervised learning algorithms.

In the first tests the performance was not entirely satisfactory, although promising developments were observed. Afterwards, the inclusion of a larger dataset and refinement of the configuration parameters for the WEKA algorithm began to show better results, managing to establish a sufficient number of clusters to cover all types of impacts represented by the examples in the dataset. The processing over a 151.826 instance dataset allowed to obtain an accurate model of impact typology, with 15 clusters associated to 15 different types of collision and a log likelihood of -9.41042. Binary attributes contained in the clusters proved to be disjoint with other clusters, allowing each cluster to be a depiction of a type of collision.

The experiments allow to conclude, therefore, that the dataset contains examples that can be divided into 15 different types of impact, covering all areas of the car susceptible to damage in an accident. Also, the examples allow to distinguish between different qualities of the vehicle, as the number of doors (3 or 5 doors) and the intensity of certain collision, such as the front or the rear impacts. It is considered that the acquired model is very good, being the result of a meticulous and thorough experimentation.

On a personal level, the project has proved to be very attractive, with a significant level of exigency due to the established goals. First, the development of the State of the Art section and the research needed to face the experimentation phase have been very enriching, because the scope of the project covers a fairly large spectrum. Certainly, the machinery that surrounds the auto-insurance claim system and the fight against insurance fraud is very interesting.

Regarding the experimentation and development, despite the use of data mining algorithms on such large datasets has taken many hours of processing and countless operations, the results are considered very satisfactory. This project could not have been undertaken without a thorough and accurate testing.

ANEXO B: Desglose de atributos

A continuación se presentan todos los atributos del fichero original de manera detallada:

Tipo	Nombre atributo	Descripcion general	Valor	Nota
Numérico	Instance_number	Número de instancia	Num	
	Secuencia	Identificador de siniestro	Num	
	Historia	Número de revisiones	Num	
Nominal	PT_51503	Techo Central	{0,1}	Atributos de pintura
	PT_53203	Faldón trasero Central	{0,1}	
	PT_55103	Capó Central	{0,1}	
	PT_55303	Portón trasero Central	{0,1}	
	PT_63203	Paragolpes delantero Central	{0,1}	
	PT_63303	Paragolpes trasero Central	{0,1}	
	PT_44201	Llanta Izda	{0,1}	
	PT_50101	Aleta delantera Izda	{0,1}	
	PT_51751	Estribo Izda	{0,1}	
	PT_53101	Aleta trasera Izda	{0,1}	
	PT_53201	Faldón trasero Izda	{0,1}	
	PT_57101	Puerta delantera Izda	{0,1}	
	PT_57201	Mecanismo de cierre puerta delantera Izda	{0,1}	
	PT_58101	Puerta trasera Izda	{0,1}	
	PT_66501	Retrovisor Izda	{0,1}	
	PT_44202	Llanta Dcha	{0,1}	
	PT_50102	Aleta delantera Dcha	{0,1}	
	PT_51752	Estribo Dcha	{0,1}	
	PT_53102	Aleta trasera Dcha	{0,1}	
	PT_53202	Faldón trasero Dcha	{0,1}	
	PT_57102	Puerta delantera Dcha	{0,1}	
	PT_57202	Mecanismo de cierre puerta delantera Dcha	{0,1}	
	PT_58102	Puerta trasera Dcha	{0,1}	
	PT_66502	Retrovisor Dcha	{0,1}	
	REP_51503	Techo Central	{0,1}	Atributos de reparación
	REP_53203	Faldón trasero Central	{0,1}	
	REP_55103	Capó Central	{0,1}	
	REP_55303	Portón trasero Central	{0,1}	
	REP_63203	Paragolpes delantero Central	{0,1}	
	REP_63303	Paragolpes trasero Central	{0,1}	
	REP_50101	Aleta delantera Izda	{0,1}	
	REP_51751	Estribo Izda	{0,1}	
	REP_53101	Aleta trasera Izda	{0,1}	
	REP_57101	Puerta delantera Izda	{0,1}	
	REP_58101	Puerta trasera Izda	{0,1}	
	REP_50102	Aleta delantera Dcha	{0,1}	
	REP_51752	Estribo Dcha	{0,1}	
	REP_53102	Aleta trasera Dcha	{0,1}	
	REP_57102	Puerta delantera Dcha	{0,1}	

	REP_58102	Puerta trasera Dcha	{0,1}	
	SUST_19103	Componentes del radiador Central	{0,1}	Atributos de sustitución
	SUST_42103	Elementos de suspensión trasera Central	{0,1}	
	SUST_50203	Componentes de la coraza frontal Central	{0,1}	
	SUST_55103	Capó Central	{0,1}	
	SUST_55303	Portón trasero Central	{0,1}	
	SUST_63203	Paragolpes delantero Central	{0,1}	
	SUST_63303	Paragolpes trasero Central	{0,1}	
	SUST_64103	Parabrisas Central	{0,1}	
	SUST_64353	Luneta trasera Central	{0,1}	
	SUST_64403	Luna custodia trasera Central	{0,1}	
	SUST_66103	Rejilla del radiador Central	{0,1}	
	SUST_66303	Anagrama del fabricante Central	{0,1}	
	SUST_66953	Matricula Central	{0,1}	
	SUST_68103	Airbag Central	{0,1}	
	SUST_68203	Cinturón de seguridad Central	{0,1}	
	SUST_70503	Tablero de instrumentos Central	{0,1}	
	SUST_87103	Aire acondicionado Central	{0,1}	
	SUST_92103	Limpiaparabrisas Central	{0,1}	
	SUST_19101	Componentes del radiador Izda	{0,1}	
	SUST_19201	Componentes del ventilador Izda	{0,1}	
	SUST_40101	Elementos de suspensión delantera Izda	{0,1}	
	SUST_40501	Suspensión delantera Izda	{0,1}	
	SUST_42101	Elementos de suspensión trasera Izda	{0,1}	
	SUST_44101	Neumático Izda	{0,1}	
	SUST_44201	Llanta Izda	{0,1}	
	SUST_44301	Tapacubos Izda	{0,1}	
	SUST_48101	Caja de dirección Izda	{0,1}	
	SUST_50101	Aleta delantera Izda	{0,1}	
	SUST_53101	Aleta trasera Izda	{0,1}	
	SUST_55101	Capó Izda	{0,1}	
	SUST_57101	Puerta delantera Izda	{0,1}	
	SUST_63201	Paragolpes delantero Izda	{0,1}	
	SUST_63301	Paragolpes trasero Izda	{0,1}	
	SUST_64201	Ventana de puerta delantera Izda	{0,1}	
	SUST_64401	Luna custodia trasera Izda	{0,1}	
	SUST_66101	Rejilla del radiador Izda	{0,1}	
	SUST_66201	Molduras Izda	{0,1}	
	SUST_66501	Retrovisor Izda	{0,1}	
	SUST_68101	Airbag Izda	{0,1}	
	SUST_68201	Cinturón de seguridad Izda	{0,1}	
	SUST_94101	Faro delantero Izda	{0,1}	
	SUST_94201	Faro trasero Izda	{0,1}	
	SUST_19102	Componentes del radiador Dcha	{0,1}	
	SUST_40102	Elementos de suspensión delantera Dcha	{0,1}	
	SUST_40402	Eje de transmisión Dcha	{0,1}	

	SUST_40502	Suspensión delantera Dcha	{0,1}	
	SUST_42102	Elementos de suspensión trasera Dcha	{0,1}	
	SUST_44102	Neumático Dcha	{0,1}	
	SUST_44202	Llanta Dcha	{0,1}	
	SUST_44302	Tapacubos Dcha	{0,1}	
	SUST_48102	Caja de dirección Dcha	{0,1}	
	SUST_50102	Aleta delantera Dcha	{0,1}	
	SUST_53102	Aleta trasera Dcha	{0,1}	
	SUST_55102	Capó Dcha	{0,1}	
	SUST_57102	Puerta delantera Dcha	{0,1}	
	SUST_63202	Paragolpes delantero Dcha	{0,1}	
	SUST_63302	Paragolpes trasero Dcha	{0,1}	
	SUST_64202	Ventana de puerta delantera Dcha	{0,1}	
	SUST_64402	Luna custodia trasera Dcha	{0,1}	
	SUST_66102	Rejilla del radiador Dcha	{0,1}	
	SUST_66202	Molduras Dcha	{0,1}	
	SUST_66502	Retrovisor Dcha	{0,1}	
	SUST_68202	Cinturón de seguridad Dcha	{0,1}	
	SUST_94102	Faro delantero Dcha	{0,1}	
	SUST_94202	Faro trasero Dcha	{0,1}	
Numérico	Pos_int	Número de piezas sustituidas	Num	
	Pos_mod	Número de piezas reparadas	Num	
	Tot_mo	Coste de mano de obra	Num	
	Tot_pint	Coste de pintura	Num	
	Tot_piez	Coste de piezas	Num	
	Tot_gen	Coste total general	Num	
Nominal	Cluster	Número de cluster asignado	{cluster1, cluster2, cluster3, cluster4}	Atributo de clase

Tabla 42: Descripción de los datos - Desglose de atributos

ANEXO C: Salidas de WEKA

A continuación se presentan las salidas completas del buffer de WEKA para el algoritmo J48.

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -R -N 3 -Q 5 -M 150
Relation:     SalidaBinario_clustered-weka.filters.unsupervised.instance.Randomize-
S11-weka.filters.unsupervised.attribute.Remove-R115_clustered-
weka.filters.unsupervised.attribute.Remove-R1-3-
weka.filters.unsupervised.attribute.Remove-R111-
weka.filters.unsupervised.instance.Randomize-S7-
weka.filters.unsupervised.attribute.Remove-R1-107
Instances:    27706
Attributes:   4
              Tot_mo
              Tot_pint
              Tot_piez
              Cluster
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

Tot_piez <= 21.01
|   Tot_pint <= 381.34
|   |   Tot_mo <= 283.3: cluster0 (5426.0/53.0)
|   |   Tot_mo > 283.3: cluster1 (154.0/53.0)
|   Tot_pint > 381.34
|   |   Tot_pint <= 527.73
|   |   |   Tot_mo <= 237.85: cluster0 (223.0/60.0)
|   |   |   Tot_mo > 237.85: cluster1 (152.0/6.0)
|   |   Tot_pint > 527.73: cluster1 (309.0/4.0)
Tot_piez > 21.01
|   Tot_mo <= 164.25
|   |   Tot_piez <= 437.21
|   |   |   Tot_pint <= 279.88: cluster2 (5027.0/63.0)
|   |   |   Tot_pint > 279.88
|   |   |   |   Tot_pint <= 363.43: cluster2 (259.0/106.0)
|   |   |   |   Tot_pint > 363.43: cluster1 (268.0/3.0)
|   |   Tot_piez > 437.21: cluster1 (553.0/66.0)
|   Tot_mo > 164.25
|   |   Tot_mo <= 709.59
|   |   |   Tot_piez <= 1584.08: cluster1 (4530.0/131.0)
|   |   |   Tot_piez > 1584.08: cluster3 (401.0/21.0)
|   |   Tot_mo > 709.59: cluster3 (1169.0/22.0)

Number of Leaves  :       12

Size of the tree :  23


Time taken to build model: 0.19 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       26673               96.2716 %
Incorrectly Classified Instances      1033                3.7284 %
Kappa statistic                          0.9477
```

```
Mean absolute error                      0.0317
Root mean squared error                  0.1282
Relative absolute error                  8.8723 %
Root relative squared error             30.3471 %
Coverage of cases (0.95 level)          97.8885 %
Mean rel. region size (0.95 level)      27.2685 %
Total Number of Instances            27706

=== Detailed Accuracy By Class ===

TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
0,967    0,006    0,985      0,967    0,976      0,966    0,995     0,987     cluster0
0,965    0,033    0,933      0,965    0,949      0,924    0,985     0,957     cluster1
0,963    0,010    0,974      0,963    0,969      0,956    0,990     0,979     cluster2
0,937    0,004    0,962      0,937    0,949      0,944    0,989     0,935     cluster3
Weighted Avg.
0,963    0,016    0,963      0,963    0,963      0,948    0,990     0,970

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
 8183  232   43    0 |    a = cluster0
   64 8616  160   90 |    b = cluster1
   60  230 7595    0 |    c = cluster2
    0  154    0 2279 |    d = cluster3
```

Tabla 43: Análisis de tasaciones – J48 primer modelo. Salida completa del mejor modelo J48

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.001 -M 200
Relation:     SalidaBinario_clustered-weka.filters.unsupervised.instance.Randomize-
S11-weka.filters.unsupervised.attribute.Remove-R115_clustered-
weka.filters.unsupervised.instance.Randomize-S7-
weka.filters.unsupervised.attribute.Remove-R1-3-
weka.filters.unsupervised.attribute.Remove-R111-
weka.filters.unsupervised.attribute.Normalize-S1.0-T0.0-
weka.filters.unsupervised.attribute.Remove-R106-110
Instances:    27706
Attributes:   106
              [list of attributes omitted]
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

SUST_68101 = 0
| SUST_19103 = 0
| | SUST_50203 = 0
| | | PT_53102 = 0
| | | | SUST_44201 = 0
| | | | | SUST_53101 = 0
| | | | | | SUST_40102 = 0
| | | | | | | SUST_50101 = 0
| | | | | | | | SUST_63303 = 0
| | | | | | | | | SUST_94102 = 0
| | | | | | | | | | SUST_57102 = 0
| | | | | | | | | | | SUST_57101 = 0
| | | | | | | | | | | | SUST_50102 = 0
| | | | | | | | | | | | | SUST_94101 = 0
| | | | | | | | | | | | | | SUST_66501 = 0
```

```
| | | | | | | | | | | | | | | SUST_63203 = 0
| | | | | | | | | | | | | | | | SUST_44301 = 0
| | | | | | | | | | | | | | | | | SUST_94201 = 0
| | | | | | | | | | | | | | | | | | PT_50101 = 0
| | | | | | | | | | | | | | | | | | | PT_63303 = 0
| | | | | | | | | | | | | | | | | | | | PT_63203 = 0
| | | | | | | | | | | | | | | | | | | | | PT_51503 = 0
| | | | | | | | | | | | | | | | | | | | | | PT_53101 = 0
| | | | | | | | | | | | | | | | | | | | | | | SUST_66202 = 0
| | | | | | | | | | | | | | | | | | | | | | | |SUST_66201 = 0: cluster2 (3117.0/621.0)
| | | | | | | | | | | | | | | | | | | | | | | |SUST_66201 = 1: cluster0 (378.0/100.0)
| | | | | | | | | | | | | | | | | | | | | | | SUST_66202 = 1: cluster0 (513.0/151.0)
| | | | | | | | | | | | | | | | | | | | | | | PT_53101 = 1: cluster0 (502.0/103.0)
| | | | | | | | | | | | | | | | | | | | | | PT_51503 = 1: cluster0 (407.0/71.0)
| | | | | | | | | | | | | | | | | | | | | | PT_63203 = 1: cluster0 (1365.0/203.0)
| | | | | | | | | | | | | | | | | | | | | | PT_63303 = 1: cluster0 (2735.0/496.0)
| | | | | | | | | | | | | | | | | | | | | PT_50101 = 1: cluster0 (1601.0/423.0)
| | | | | | | | | | | | | | | | | | | | | SUST_94201 = 1: cluster2 (360.0/182.0)
| | | | | | | | | | | | | | | | | | | | | SUST_44301 = 1: cluster2 (224.0/99.0)
| | | | | | | | | | | | | | | | | | | | | SUST_63203 = 1: cluster2 (795.0/247.0)
| | | | | | | | | | | | | | | | | | | | | SUST_66501 = 1: cluster2 (394.0/128.0)
| | | | | | | | | | | | | | | | | | | | | SUST_94101 = 1: cluster2 (429.0/133.0)
| | | | | | | | | | | | | | | | | | | | | SUST_50102 = 1: cluster2 (364.0/107.0)
| | | | | | | | | | | | | | | | | | | | | SUST_57101 = 1: cluster1 (319.0/160.0)
| | | | | | | | | | | | | | | | | | | | | SUST_57102 = 1: cluster1 (219.0/94.0)
| | | | | | | | | | | | | | | | | | | | | SUST_94102 = 1
| | | | | | | | | | | | | | | | | | | | | | PT_55103 = 0: cluster2 (421.0/158.0)
| | | | | | | | | | | | | | | | | | | | | | PT_55103 = 1: cluster1 (215.0/70.0)
| | | | | | | | | | | | | | | | | | | | | SUST_63303 = 1
| | | | | | | | | | | | | | | | | | | | | | SUST_66303 = 0: cluster2 (1888.0/493.0)
| | | | | | | | | | | | | | | | | | | | | | SUST_66303 = 1
| | | | | | | | | | | | | | | | | | | | | | PT_53203 = 0: cluster2 (299.0/116.0)
| | | | | | | | | | | | | | | | | | | | | | PT_53203 = 1: cluster1 (303.0/68.0)
| | | | | | | | | | | | | | | | | | | | | SUST_50101 = 1
| | | | | | | | | | | | | | | | | | | | | | SUST_66201 = 0
| | | | | | | | | | | | | | | | | | | | | | SUST_63203 = 0: cluster2 (420.0/109.0)
| | | | | | | | | | | | | | | | | | | | | | SUST_63203 = 1: cluster1 (214.0/92.0)
| | | | | | | | | | | | | | | | | | | | | | SUST_66201 = 1: cluster1 (413.0/122.0)
| | | | | | | | | | | | | | | | | | | | | SUST_40102 = 1: cluster1 (292.0/86.0)
| | | | | | | | | | | | | | | | | | | | | SUST_53101 = 1: cluster1 (371.0/155.0)
| | | | | | | | | | | | | | | | | | | | SUST_44201 = 1: cluster1 (325.0/168.0)
| | | | | | | | | | | | | | | | | | | PT_53102 = 1
| | | | | | | | | | | | | | | | | | | PT_53101 = 0
| | | | | | | | | | | | | | | | | | | | SUST_53102 = 0
| | | | | | | | | | | | | | | | | | | | SUST_50102 = 0
| | | | | | | | | | | | | | | | | | | | SUST_63303 = 0
| | | | | | | | | | | | | | | | | | | | REP_63203 = 0
| | | | | | | | | | | | | | | | | | | | PT_63203 = 0: cluster0 (2160.0/749.0)
| | | | | | | | | | | | | | | | | | | | PT_63203 = 1: cluster1 (244.0/90.0)
| | | | | | | | | | | | | | | | | | | | REP_63203 = 1: cluster1 (229.0/72.0)
| | | | | | | | | | | | | | | | | | | | SUST_63303 = 1: cluster1 (313.0/136.0)
| | | | | | | | | | | | | | | | | | | | SUST_50102 = 1: cluster1 (222.0/47.0)
| | | | | | | | | | | | | | | | | | | | SUST_53102 = 1: cluster1 (281.0/121.0)
| | | | | | | | | | | | | | | | | | | | PT_53101 = 1: cluster1 (2519.0/427.0)
| | | | | | | | | | | | | | | | | | | SUST_50203 = 1
| | | | | | | | | | | | | | | | | | | | PT_57102 = 0: cluster1 (1397.0/320.0)
| | | | | | | | | | | | | | | | | | | | PT_57102 = 1: cluster3 (207.0/86.0)
| | | | | | | | | | | | | | | | | | | SUST_19103 = 1
| | | | | | | | | | | | | | | | | | | | SUST_19201 = 0
| | | | | | | | | | | | | | | | | | | | SUST_50102 = 0: cluster1 (281.0/110.0)
| | | | | | | | | | | | | | | | | | | | SUST_50102 = 1: cluster3 (228.0/70.0)
| | | | | | | | | | | | | | | | | | | | SUST_19201 = 1: cluster3 (367.0/52.0)
SUST_68101 = 1: cluster3 (375.0/6.0)
```

```
Number of Leaves  :        40

Size of the tree :  79


Time taken to build model: 4.25 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       19988               72.1432 %
Incorrectly Classified Instances      7718               27.8568 %
Kappa statistic                          0.6056
Mean absolute error                      0.2033
Root mean squared error                  0.3195
Relative absolute error                 56.9409 %
Root relative squared error             75.6112 %
Coverage of cases (0.95 level)          98.2892 %
Mean rel. region size (0.95 level)      66.7256 %
Total Number of Instances            27706

=== Detailed Accuracy By Class ===

TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
0,871    0,134    0,741      0,871    0,801      0,708    0,924     0,797     cluster0
0,615    0,123    0,704      0,615    0,657      0,511    0,829     0,696     cluster1
0,784    0,128    0,709      0,784    0,745      0,637    0,891     0,729     cluster2
0,390    0,012    0,758      0,390    0,515      0,515    0,903     0,617     cluster3
Weighted Avg.
0,721    0,118    0,722      0,721    0,713      0,608    0,882     0,729

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
 7363  206  889    0 |    a = cluster0
 1621 5495 1538  276 |    b = cluster1
  817  860 6180   28 |    c = cluster2
  133 1244  106  950 |    d = cluster3
```

Tabla 44: Análisis de tasaciones – J48 binarios. Salida completa del mejor modelo J48

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.001 -M 200
Relation:     SalidaBinario_clustered-weka.filters.unsupervised.instance.Randomize-
S11-weka.filters.unsupervised.attribute.Remove-R115_clustered-
weka.filters.unsupervised.attribute.Remove-R1-3-
weka.filters.unsupervised.attribute.Remove-R108-111-
weka.filters.unsupervised.instance.Randomize-S7
Instances:    27706
Attributes:   108
              [list of attributes omitted]
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
------------------

Pos_int <= 12
| SUST_50203 = 0
| | SUST_64103 = 0
```

```
| | | SUST_57102 = 0
| | | | SUST_57101 = 0
| | | | | SUST_50101 = 0
| | | | | | SUST_63303 = 0
| | | | | | | SUST_50102 = 0
| | | | | | | | SUST_94101 = 0
| | | | | | | | | SUST_94102 = 0
| | | | | | | | | | SUST_66501 = 0
| | | | | | | | | | | SUST_63203 = 0
| | | | | | | | | | | | SUST_94201 = 0
| | | | | | | | | | | | | Pos_mod <= 0
| | | | | | | | | | | | | | SUST_66202 = 0
| | | | | | | | | | | | | | | PT_53101 = 0
| | | | | | | | | | | | | | | | SUST_66201 = 0
| | | | | | | | | | | | | | | | | Pos_int <= 5: cluster2 (1128.0/421.0)
| | | | | | | | | | | | | | | | | Pos_int > 5: cluster0 (285.0/167.0)
| | | | | | | | | | | | | | | | SUST_66201 = 1: cluster0 (208.0/53.0)
| | | | | | | | | | | | | | | PT_53101 = 1: cluster0 (279.0/73.0)
| | | | | | | | | | | | | | SUST_66202 = 1: cluster0 (458.0/113.0)
| | | | | | | | | | | | | Pos_mod > 0: cluster0 (7363.0/1436.0)
| | | | | | | | | | | | SUST_94201 = 1: cluster2 (295.0/129.0)
| | | | | | | | | | | SUST_63203 = 1: cluster2 (703.0/170.0)
| | | | | | | | | | SUST_66501 = 1: cluster2 (298.0/44.0)
| | | | | | | | | SUST_94102 = 1: cluster2 (293.0/64.0)
| | | | | | | | SUST_94101 = 1: cluster2 (406.0/94.0)
| | | | | | | SUST_50102 = 1
| | | | | | | | Pos_int <= 7: cluster2 (220.0/19.0)
| | | | | | | | Pos_int > 7: cluster1 (206.0/97.0)
| | | | | | SUST_63303 = 1: cluster2 (2152.0/461.0)
| | | | | SUST_50101 = 1: cluster2 (610.0/160.0)
| | | | SUST_57101 = 1: cluster2 (313.0/157.0)
| | | SUST_57102 = 1: cluster1 (274.0/97.0)
| | SUST_64103 = 1: cluster2 (1561.0/61.0)
| SUST_50203 = 1: cluster1 (377.0/72.0)
Pos_int > 12
| SUST_68101 = 0
| | SUST_19201 = 0
| | | SUST_42103 = 0
| | | | Pos_int <= 22: cluster1 (4699.0/1294.0)
| | | | Pos_int > 22
| | | | | SUST_40102 = 0
| | | | | | Pos_int <= 45: cluster1 (3366.0/630.0)
| | | | | | Pos_int > 45
| | | | | | | SUST_50203 = 0
| | | | | | | | Pos_int <= 58: cluster1 (473.0/166.0)
| | | | | | | | Pos_int > 58: cluster3 (208.0/52.0)
| | | | | | | SUST_50203 = 1: cluster3 (202.0/12.0)
| | | | | SUST_40102 = 1: cluster3 (304.0/75.0)
| | | SUST_42103 = 1: cluster3 (275.0/57.0)
| | SUST_19201 = 1: cluster3 (379.0/56.0)
| SUST_68101 = 1: cluster3 (371.0/3.0)

Number of Leaves  :        28

Size of the tree :  55


Time taken to build model: 2.83 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances       21346               77.0447 %
```

```
Incorrectly Classified Instances      6360               22.9553 %
Kappa statistic                          0.6767
Mean absolute error                      0.1775
Root mean squared error                  0.2988
Relative absolute error                 49.717  %
Root relative squared error             70.7101 %
Coverage of cases (0.95 level)          98.6285 %
Mean rel. region size (0.95 level)      63.5747 %
Total Number of Instances            27706

=== Detailed Accuracy By Class ===

TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
0,792    0,097    0,782      0,792    0,787      0,692    0,911     0,757     cluster0
0,775    0,122    0,752      0,775    0,763      0,648    0,874     0,746     cluster1
0,782    0,094    0,768      0,782    0,775      0,684    0,905     0,804     cluster2
0,641    0,014    0,820      0,641    0,720      0,702    0,954     0,763     cluster3
Weighted Avg.
0,770    0,097    0,772      0,770    0,770      0,677    0,901     0,767

=== Confusion Matrix ===

    a    b    c    d   <-- classified as
 6699  880  879    0 |    a = cluster0
  698 6922  967  343 |    b = cluster1
 1150  570 6165    0 |    c = cluster2
   22  835   16 1560 |    d = cluster3
```

Tabla 45: Análisis de tasaciones – J48 binarios y piezas. Salida completa del mejor modelo J48

ANEXO D: Árboles de clasificación J48

A continuación se presentan los árboles generados por el algoritmo de WEKA J48.

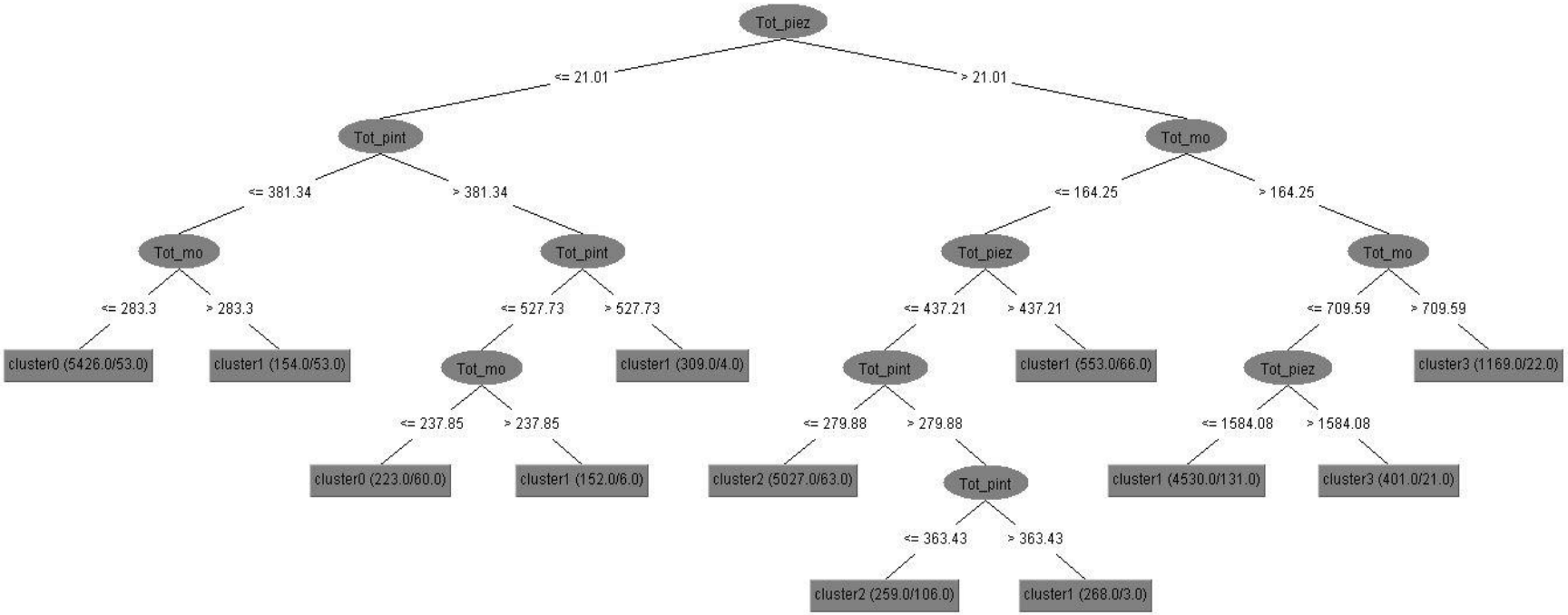


Ilustración 23: Análisis de tasaciones – J48 primer modelo. Árbol generado por el mejor modelo

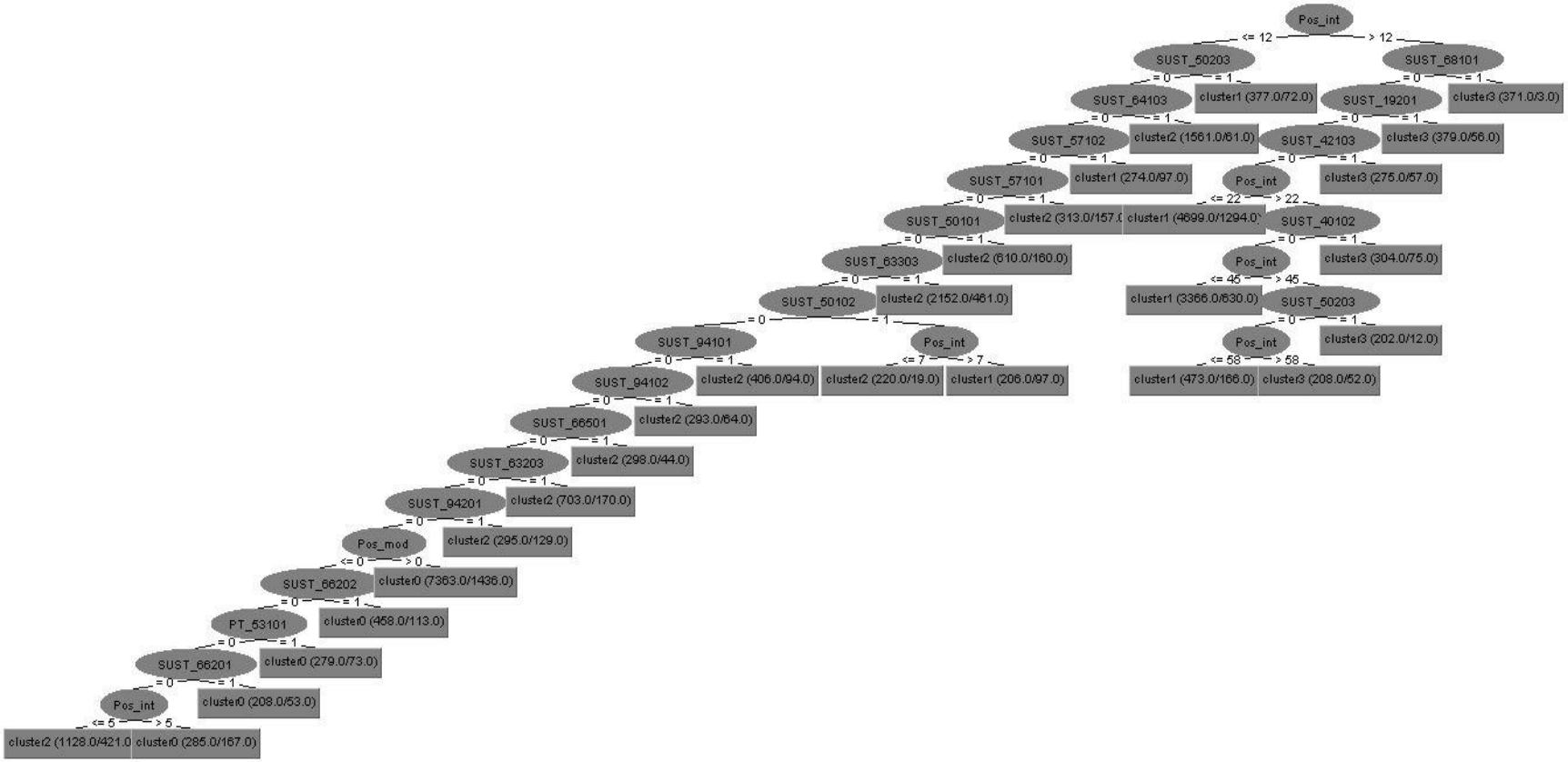


Ilustración 24: Análisis de tasaciones – J48 binarios y piezas. Árbol generado por el mejor modelo

ANEXO E: Planificación

En esta sección se presenta la planificación del proyecto:

	Nombre	Duración	Inicio	Fin
1	Fase Inicial	8d	16/02/2015	23/02/2015
2	Investigación	30d	24/02/2015	25/03/2015
3	Experimentación	30d	26/03/2015	24/04/2015
4	Elaboración de la memoria	30d	25/04/2015	23/05/2015
5	Revisión General	7d	24/05/2015	30/05/2015
6	Contratiempos	15d	31/05/2015	14/06/2015
7	Presentación	1.6d	02/07/2015	03/07/2015

Ilustración 25: Planificación inicial

	Nombre	Duración	Inicio	Fin
1	**Fase Inicial**	**10d**	**16/02/2015**	**25/02/2015**
2	Proposición de Proyecto	1d	16/02/2015	16/02/2015
3	Preparación Entorno	3d	16/02/2015	18/02/2015
4	Familiarización y definición de objetivos	7d	19/02/2015	25/02/2015
5	**Investigación de documentación**	**21d**	**26/02/2015**	**18/03/2015**
6	Búsqueda de bibliografía	2d	26/02/2015	27/02/2015
7	Documentación sobre data mining	5d	28/02/2015	04/03/2015
8	Documentación sobre peritación y fraude	7d	07/03/2015	13/03/2015
9	Investigación sobre proyectos similares	5d	14/03/2015	18/03/2015
10	**Experimentación**	**69d**	**19/03/2015**	**26/05/2015**
11	Depuración de datasets	2d	19/03/2015	20/03/2015
12	Análisis y pruebas iniciales	15d	21/03/2015	04/04/2015
13	Análisis de severidad	30d	06/04/2015	05/05/2015
14	Auditoría de daños	27d	30/04/2015	26/05/2015
15	**Elaboración de la memoria**	**98d**	**26/02/2015**	**03/06/2015**
16	Estructura general del documento	1d	26/02/2015	26/02/2015
17	Introducción, contexto, objetivos	3d	02/03/2015	04/03/2015
18	Estado del arte	30d	19/03/2015	17/04/2015
19	Diseño de la solución	7d	04/05/2015	10/05/2015
20	Resultados y evaluación	7d	27/05/2015	02/06/2015
21	Conclusiones	1d	03/06/2015	03/06/2015
22	Revisión General	3d	10/06/2015	12/06/2015
23	Presentación	1d	09/07/2015	09/07/2015

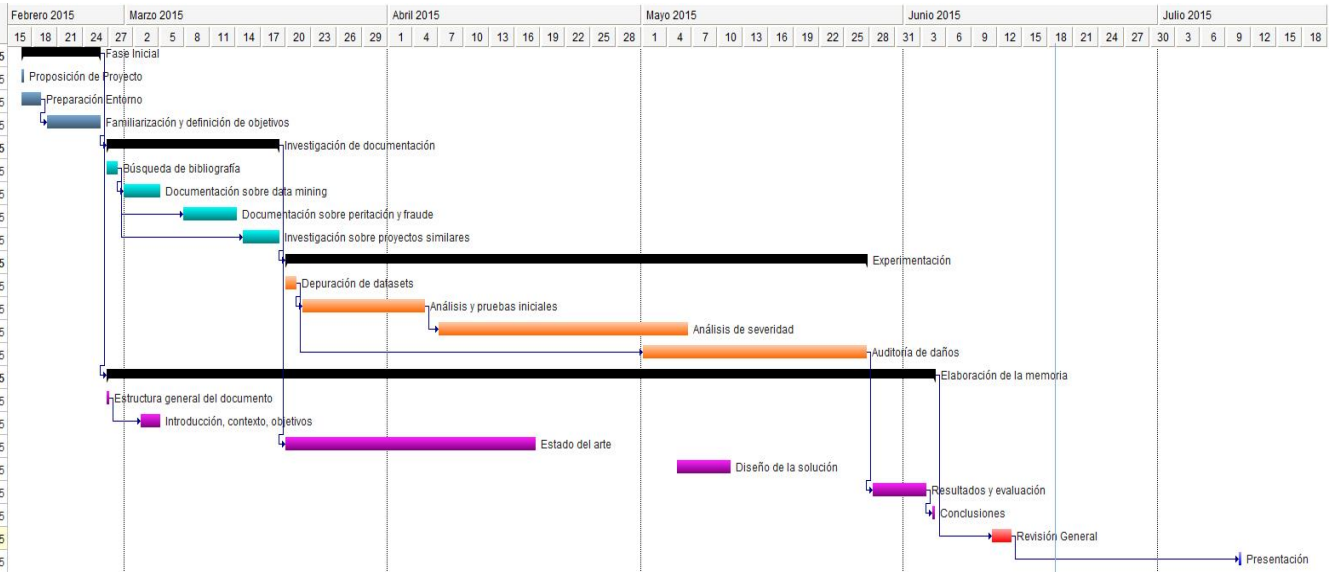


Ilustración 26: Planificación final